

# TUsersCS para Delphi 3/4

## Manual del Usuario

### por Tools&Comps

Tools&Comps Home Page: [www.gtc.com.br/~jgkrs/tandco.htm](http://www.gtc.com.br/~jgkrs/tandco.htm)

Tools&Comps e-mail: [tandco@gtc.com.br](mailto:tandco@gtc.com.br)

**VISIÓN GENERAL.....3**

**INSTALANDO EL COMPONENTE TUSERSCS.....4**

**COMO FUNCIONA EL COMPONENTE TUSERSCS.....4**

1. CREANDO LAS TABLAS DE USUARIOS.....4

2. USANDO EL TUSERSCS DENTRO DEL IDE DEL DELPHI.....4

3. OTRAS PROPIEDADES DEL TUSERSCS.....5

4. PROPIEDADES DE LA CLASE TUSER.....5

5. MÉTODOS DEL COMPONENTE TUSERSCS.....6

**SOPORTE A OTROS IDIOMAS.....6**

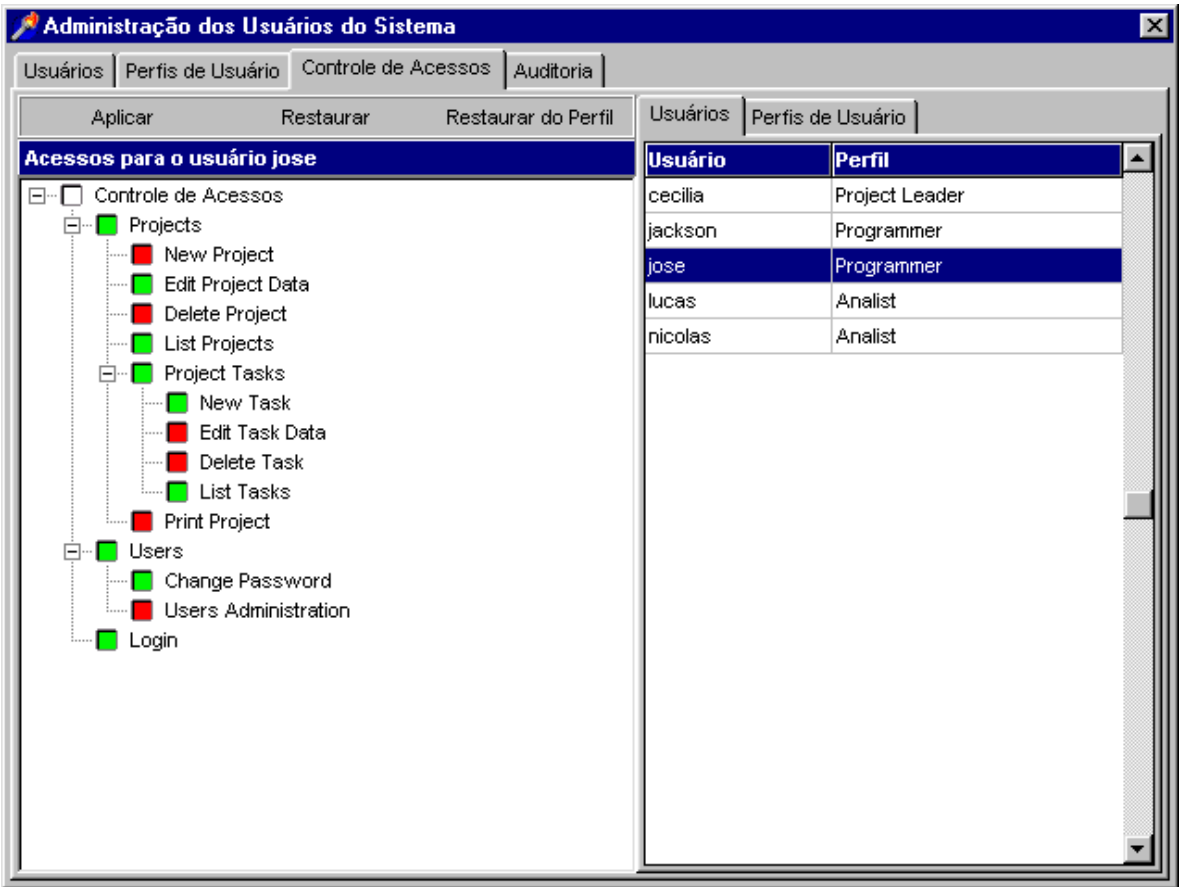
**REGISTRANDO EL COMPONENTE TUSERSCS.....6**

# VISIÓN GENERAL

**TUsersCS** (versión **C**lient/**S**erver) es un componente que torna simples implementar el control de seguridad en programas escritos en Delphi. Con una pequeña configuración a nivel de componente en tiempo de desing, su aplicación estará lista para controlar el acceso de sus usuários. Con el **TUsersCS**, usted no necesita escribir ninguna línea de código para implementar el control de usuarios. Sólo lo conecte al componente **TMenú** de su formulario principal e deje todo por cuenta del componente **TUsersCS**.

A través de la conexión com el menú del formulário principal de su aplicación, el **TUsersCS** permite que el administrador de seguridad de su sistema conceda permisos de acceso al usuario a cada item de menú. Todos los accesos del usuario son exhibidos en un árbol que imita la hierarquia de menús de su programa. El administrador sólo debe clicar en el item desejado para conceder o retirar el acceso del usuáριο para ese item. Cuando el usuario logar en el sistema, el **TUsersCS** si encarga de desabilitar o tornar invisible (de acuerdo con el perfil ao qual el usuario pertenece) los items de menú los cuales el usuario no tiene acceso.

Además, El componente **TUsersCS** cuenta con un módulo de auditoria, que permite averiguar la utilización de los programas por los usuarios.



El modulo de administración de usuarios es muy simples de usar, principalmente para el usuario final de sus aplicaciones. En este modulo, el administrador puede:

- crear usuarios;
- crear perfiles de usuario;
- asociar usuarios a uno de los perfiles;
- conceder permisos de acceso a los usuarios (individualmente) e a los perfiles, visualmente;
- actualizar los permisos de acceso de todos los usuarios que pertenecen a un perfil;
- cambiar datos y contraseña de usuario;
- Obligar el cambio de contraseña de usuarios que pertenecen a un perfil a cada intervalo de días especificado
- poner un usuario o un perfil de usuario en modo de auditoria y averiguar la utilización de los programas por los usuarios, entre otras acciones.

El componente **TUsersCS** puede ser usado con cualquier SGBD del mercado que soporte el standard ANSI-SQL y que estea accessible por el Delphi usando **BDE** o **ODBC**.

Otra importante característica del **TUsersCS** es que las tablas de control de usuarios pueden ser compartidas por diversas aplicaciones al mismo tiempo, no se haciendo necesario reingresar los usuários en todas las aplicaciones.

**TUsersCS** es implementado en Object Pascal puro, inclusive el módulo de administración de usuarios, que es implementado juntamente con el componente. No se hacen necesarias librerías externas ni programas adicionales que necesiten ser enviados juntamente con su programa en Delphi para que el componente funcione correctamente. Usted sólo debe enviar las tablas de usuarios juntamente con su sistema.

## INSTALANDO EL COMPONENTE TUSERSCS

---

Para instalar el componente **TUsersCS**:

- abra el archivo ToolsAndComps\UsersCS\Delphi3Trial\uCSD3.dpk, a través del menú File|Open de su Delphi y clique en el botón **Install**.
- En el IDE del Delphi, abra Tools|Enviroment Options|Library|Library Path, y agregue c:\ToolsAndComps\UsersCS\Delphi3Trial, haciendo click en el botón **OK**.

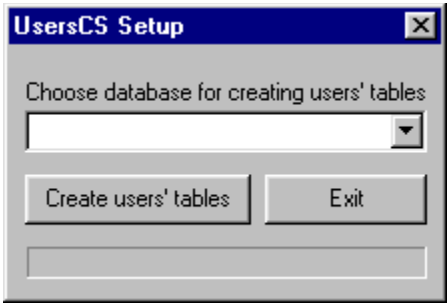
OBS: Los procedimientos son los mismos para Delphi 4.

## COMO FUNCIONA EL COMPONENTE TUSERSCS

---

### 1. Creando las tablas de usuarios

El primer paso para usar el **TUsersCS** es crear las tablas que irán almacenar las informaciones sobre los usuarios de su sistema. El componente **TUsersCS** viene acompañado de un utilitario que auxilia en la creación de esas tablas, o **UCSSetup.EXE**.



Para crear las tablas de usuarios, usted debe informar el alias(base de datos) en la qual usted desea almacenar las tablas (usted debe crear previamente un alias en el BDE o un DSN en el administrador ODBC de su computador). Despues de informar el alias(base de datos), haga click en el botón **“Create users's tables”** para que el utilitário pueda crer las tablas de acuerdo con las informaciones especificadas. Las tablas a serem creadas son:

1. USERSCS
2. USERINFO0
3. PROFILES
4. PROGS
5. UACCESS
6. PACCESS
7. HISTORY
8. AUDIT

**Dica:** Usted puede criar las tablas de usuarios en la misma base de datos que su aplicación va a usar o también puede crear una base de datos específica para el control de usuarios.

**Nota:** Por el facto de ser el desarrollador quién elije el SGBD aonde van a ser creadas las tablas de usuarios, la seguridad de esas tablas debe ser la misma que las otras tablas de su sistema.

Cuando el utilitario **UCSSetup.EXE** crea las tablas de usuarios, automáticamente inserta un registro con o nombre de usuario '**master**', con contraseña también '**master**' e total acceso al sistema. Ese es el usuario que irá gerenciar el sistema de seguridad de sus programas, no siendo posible deletar ese usuario, siendo apenas posible cambiar su contraseña (lo que se recomienda).

**Nota:** Para una mejor seguridad del sistema, las contraseñas de todos los usuarios ingresados son criptografadas.

## 2. Usando el TUsersCS dentro del IDE del Delphi

Para usar el TUsersCS en sus programas, en la paleta de componentes, seleccione la guía **ToolsAndComps**, después adicione o componente al formulario principal de su programa.

Para que el componente funcione correctamente, usted debe ingresar inicialmente el valor de las siguientes propiedades en el Object Inspector:

*AppKey: String[20]*

Clave para la aplicación. Esa propiedad és obligatória y permite que las tablas de usuarios seam compartidas por várias aplicações. Esa propriedade va a funcionar como a clave da aplicación actual y debe ser unica enre las aplicaciones que van a compartir la misma base de datos de usuarios. Note que el componente no puede verificar si la clave és unica. El propio desarrollador debe tener los cuidados a respecto de la clave.

*DatabaseName: String*

Indica el alias ao qual el componente **TUsersCS** va a ser conectado para abrir las tablas de control de usuarios.

**IMPORTANTE:** A través del alias informado como parâmetro, el componente va a intentar abrir la base de datos de usuarios(Creada a través del utilitario **UCSSetup**) usando las propiedades del alias, incluyendo el usuario y la contraseña usada por su aplicación para abrir su base de datos. Note que el componente debe estar "autorizado" a accesar la base de datos de usuarios. Esa autorización puede ser echa de la siguiente forma:

1. Solicite a su DBA para crear un usuario en su banco de datos e que él, y solamente él, tenga acceso a las tablas de control de usuarios.
2. Agregue un componente TDatabase a su formulario principal y conecteo a la base de datos aonde están las tablas de control de usuarios.
3. En la propiedad Params del TDatabase creado, agregue las siguientes líneas:  
USER NAME="Nombre Del Usuario Creado Por el DBA"  
PASSWORD="Password Del Usuario Creado"
4. Cambie el valor de la propiedad LoginPrompt para FALSE
5. Conecte el componente TUsersCS de su formulario principal al componente TDatabase creado

Vea los ejemplos que acompanham el componente

*Menu: TMenu*

Indica el componente de menu lo qual el **TUsersCS** va a controlar. Se su formulário ya tiene un componente TMenu, el **TUsersCS** si conecta automaticamente a él.

**Importante:** Note que el alias (propiedade DatabaseName) debe ser el mismo que usted forneció cuando creó las tablas de usuarios usando el utilitário **UCSSetup.EXE**. Sin los valores correctos, el componente no pondrá funcionar correctamente.

## 3. Otras propiedades del TUsersCS

*AutoLogin: Boolean*

Especifica si el componente va a llamar la función de Login en el sistema automáticamente. Si AutoLogin es **True**, el componente llama la función de Login cuando la aplicación es iniciada. Si el usuario no consigue logar en la aplicación después de 3 tentativas, el componente termina la a aplicación. Si AutoLogin es **False** la función de Login debe ser llamada manualmente.

*ActualUser: TUser*

Esa propiedad contiene datos sobre el usuario actualmente logado en el sistema.

#### 4. Propiedades de la Clase TUser

- UserName: String

Nombre del usuario logado
- RealName: String

Nombre real del usuario logado
- UserId: Integer

Número de identificación del usuario logado
- Password: String

Contraseña del usuario logado

**Nota:** El componente mantiene un historico de contraseñas de todos los usuários, de forma que este sea obligado a usar contraseñas distintas, cada vez que cambia su contraseña. Para que uno usuario pueda reutilizar una contraseña, el administrador debe remover la contraseñas del historico de contraseñas del usuáριο, en el módulo de administración.
- LastPwdChange: TDateTime

Fecha del último cambio de contraseña del usuario
- ExpirationDate: TDateTime

Fecha de expiración del usuario dentro del sistema. A partir de esa fecha el usuario no puede mas logar en el sistema. Para volver a usar el sistema, el administrador de seguridad debe hacer-lo activo e cambiar su fecha de expiración. Esa propiedad sólo hace sentido si la propiedad *UserExpire* es TRUE.

**Importante:** Observe que este valor está vinculado a aplicación, o sea, un usuario tiene la fecha de expiración distinta para cada aplicación.
- UserExpire: Boolean

Indica si el usuáριο expira dentro del sistema.

**Importante:** Observe que este valor está vinculado a aplicación, o sea, un usuario puede expirar o no, de acuerdo con la aplicación.
- UserActive: Boolean

Indica si el usuario está activo, o sea, si el usuáριο es válido dentro del sistema o ha expirado.

**Importante:** Observe que este valor está vinculado a aplicación, o sea, un usuario puede estar activo o no, de acuerdo con la aplicación.
- Profile: String

Retorna el nombre del perfil al qual el usuario pertenece.
- ProfileId: Integer

Retorna el identificador del perfil al qual el usuario pertenece.
- AuditMode: Boolean

Indica si el usuario está en modo de auditoria o no.

**Nota:** Todas las propiedades de la clase **TUser** son sólo lectura.

#### 5. Métodos del componente TUsersCS

- Login: boolean

Ese método debe ser usado para llamar la pantalla de Login del sistema. Si después de 3 tentativas, el usuario no consigue logar en el sistema, esa función booleana va a retornar **False**. Para un login bien sucedido, esa función va a retornar **True**. La llamada a esa función se hace necesaria cuando la propiedad *AutoLogin* es **False**. En esta situación, el desarrollador debe escribir el código necesario para que la aplicación no pueda ser usada por un usuario no autorizado. Vea los ejemplos.
- ChangeUserPassword

Ese método permite al usuario logado efectuar el cambio de su contraseña. Debe ser llamado manualmente por el Desarrollador.
- UsersAdm

Llama el módulo de administración de usuarios del sistema. Debe ser llamado manualmente por el Desarrollador.

### SOPORTE A OTROS IDIOMAS

El componente **TUsersCS** fue proyectado para que el dessarrollador pueda modificar fácilmente el idioma usado por el componente. La versión actual tiene traducciones de

todas sus strings de interface con el usuario en portugués, español e inglés. Para cada uno de los idiomas citados, el paquete del componente **TUsersCS** viene con el archivo de recursos ya compilado (UsersCSRes.RES), el archivo fuente de la STRING TABLE (UsersCSRes.RC) y el archivo con las constantes (UsersCSRes.PAS). Para compilar su programa para uno de los idiomas citados, basta incluir en su proyecto, el archivo UsersCSRes.Pas que está en el directorio del idioma deseado.

Si usted quiere usar el componente en alemán, por ejemplo, siga los siguientes pasos:

1. Dentro del directorio ToolsAndComps\UsersCS\Resources, cree un directorio para el idioma deseado.
2. Copie todos los archivos del directorio de alguno de los idiomas para el nuevo directorio.
3. Traduzca el archivo UsersCSRes.RC para el idioma deseado.
4. Compile el nuevo archivo UsersCSRes.RC utilizando el utilitario BRCC32.EXE, que está en el directorio BIN de su instalación Delphi. Consulte su documentación del Delphi a respecto del uso del BRCC32.EXE.
5. Agregue en su proyecto, el archivo UsersCSRes.Pas que está en el directorio del nuevo idioma. El nuevo archivo UsersCSRes.RES generado también debe estar en ese directorio.
6. Recompile su proyecto.

## **REGISTRANDO EL COMPONENTE TUSERSCS**

---

Visite nuestra home page para obtener informaciones actualizadas sobre como registrar el componente TUsersCS v1.0.