

Usuarios de bases de datos: una percepción errónea en el aula de clases

Homero Jiménez Pérez
Universidad de la Sierra Sur
jphomero@yahoo.com.mx

Lirio Ruiz Guerra
Universidad de la Sierra Sur
liriorg@gmail.com

Resumen

En la enseñanza de la asignatura bases de datos existen algunos conceptos que generan confusión entre los estudiantes al momento de implementarlos. Esto se puede convertir en un problema conforme avanza su formación, al grado de realizar proyectos finales de semestre con los mismos errores. La temática que se abarcará en el presente artículo es la perspectiva errónea que los estudiantes tienen sobre el término “usuario” en las bases de datos. Inicialmente, se piensa que la información de un usuario va incluida en el contenido de una base de datos. Sin embargo, esta perspectiva está totalmente alejada de su verdadero significado.

Esta errónea interpretación que se genera en el aula de clases es importante erradicarla porque los estudiantes siguen alimentando esa idea e identifican correctamente el término “usuario” ya avanzado el curso de bases de datos, o en casos más críticos, ya concluido. Las aplicaciones que se desarrollan desde este enfoque erróneo derivan en problemas de seguridad en la aplicación y en la propia base de datos.

Para enfrentar esta problemática, se debe aclarar desde el principio el concepto de “usuario” y las implicaciones que tiene en el desarrollo de sistemas. Se muestra una aplicación que involucra usuarios reales de la base de datos, así como los derechos que tienen dentro de la misma.

Con lo anterior, se consigue la interpretación correcta del concepto, evitando de esta manera errores en el desarrollo de aplicaciones de software.

Palabras clave: usuario, bases de datos, enseñanza, aprendizaje.

Introducción

En el aula de clases es muy frecuente que existan problemas de comunicación entre docentes y alumnos. El docente debe tener las habilidades necesarias para facilitar el conocimiento a sus estudiantes de forma tal que asimilen perfectamente lo enseñado. Cuando nosotros no tenemos un dominio completo de esas habilidades, se puede generar en los estudiantes una mala interpretación de los conceptos que van llevando a lo largo de su formación académica. Es importante evitar las erróneas interpretaciones mediante una explicación detallada de cada uno de los temas abordados en el aula de clases. El propósito del presente documento es mostrar como parte de la experiencia, la manera en que estudiantes de la asignatura Bases de Datos confunden de una manera muy común el concepto de “usuario de bases de datos”. Cuando no se trabaja adecuadamente en la comprensión de este tema, el estudiante lleva esta interpretación errónea hasta tareas fundamentales como el desarrollo e implementación de aplicaciones de software.

Dentro del área de desarrollo de software, cuando es asignado un proyecto para satisfacer una determinada necesidad de manejo de información, este debe cumplir con las expectativas de solución. En la mayoría de las aplicaciones, el acceso a la información por parte de los usuarios es restringido. Dicha restricción es importante porque implica temas de seguridad de la información. Cuando se hace una mala distribución de usuarios y sus derechos de acceso, la solución planteada es carente de calidad y, consecuentemente, genera problemas mucho más graves que aquellos que se pretenden solucionar.

Desarrollo de contenidos

Antecedentes

En la Universidad de la Sierra Sur, dentro de la licenciatura en informática se imparten asignaturas que tienen que ver con aplicaciones de desarrollo de software. Estas asignaturas son programación estructurada, estructuras de datos, paradigmas de programación I y II, diseño web, bases de datos I y II, ingeniería de software I y II. Con estos conocimientos generan aplicaciones que tienen un impacto en su propio contexto, como la misma Universidad, su municipio, etcétera.

Dentro de nuestra experiencia en el aula, hemos notado que cuando se asigna a un grupo de estudiantes la realización de un proyecto de desarrollo que involucre bases de datos, frecuentemente se confunde la asignación de derechos de accesos a los usuarios que utilizarán el sistema.

Como ejemplo podemos mencionar una aplicación de gestión de calificaciones que se asignó como proyecto de desarrollo para un grupo de 6º semestre de la asignatura ingeniería de software. Al momento de asignar derechos a usuarios para la consulta de la información, se hacía de manera errónea lo que provocaba que la aplicación no tuviera los detalles mínimos de seguridad. Ante esto, cada usuario que ingresaba al sistema lo hacía con todos los privilegios sobre la base de datos. Evidentemente, los usuarios finales y desarrolladores no tenían el conocimiento del gran hueco de seguridad que se había generado.

Como el caso anterior, existen otros en los que esta confusión se ha presentado. Entre las causas podemos citar que concretamente no existe en los planes de estudio de las asignaturas de bases de datos, un capítulo o tema que aborde de manera detallada la asignación de privilegios a los usuarios, mucho menos cómo implementarlo en un sistema de software. Es responsabilidad de cada docente insertar esta temática durante el desarrollo de los cursos.

Cuando nosotros –en el papel de usuario final– accedemos a un sistema para consultar información, lo hacemos con las restricciones adecuadas. Solamente se permite la consulta y edición de información determinada. En otras palabras, formamos parte de un grupo de usuarios restringidos. Por ejemplo, simplemente cuando realizamos el pago de la cuenta del supermercado, el cajero se convierte en el usuario final de un sistema de información que incluye –seguramente– el manejo de una base de datos. La función que tiene habilitada es únicamente la de consultar información sobre los productos que los clientes desean adquirir y, a partir de ahí, generar el ticket de venta. Esto es relevante debido a que el usuario tiene bien definidas las actividades que puede hacer sobre el sistema y la base de datos.

Omisiones y errores en la definición de roles en el sistema provocan grandes daños, que van desde la alteración de los datos hasta la completa pérdida de la información. Por lo tanto, la seguridad implica garantizar que los usuarios están autorizados para llevar a cabo lo que tratan de hacer sobre la base de datos.

Elección de la temática

En el presente trabajo, ubicamos como referencia a alumnos de la licenciatura en informática que cursan asignaturas de 6º y 7º semestre. En esta etapa de su carrera, han tenido oportunidad de cursar estudios de programación estructurada, estructuras de datos, paradigmas de programación I, diseño web, bases de datos I, programación de sistemas, paradigmas de programación II, ingeniería de software I, bases de datos II, ingeniería de software II. Tienen las herramientas necesarias para desarrollar una aplicación de software cuyo objetivo es enfocado a resolver una necesidad de manejo de información.

La temática a tratar se ha seleccionado a partir de una serie de deficiencias detectadas en las aplicaciones de software que desarrollan los estudiantes. Con los conocimientos que previamente han adquirido, logran desarrollar proyectos confiables en el sentido de cumplir las necesidades que le dan origen. Sin embargo, en varios casos se presenta un aspecto que no se considera al momento de la implementación: los privilegios de los usuarios que utilizan las aplicaciones desarrolladas.

Descripción de la temática

Cada semestre, los estudiantes de la licenciatura en informática realizan un proyecto final. Está relacionado con el contenido abordado en las asignaturas durante el transcurso del semestre. La complejidad depende del semestre y materia que estén cursando. Generalmente, es en el 5º y 6º semestre que empiezan a desarrollar aplicaciones más complejas y completas.

Con los conocimientos adquiridos en asignaturas como ingeniería de software I y II, pueden detectar con mayor exactitud las necesidades de automatización de la información. Esto repercute en que desarrollen aplicaciones que cumplen y satisfacen los requerimientos, en otras palabras, cumplen su objetivo.

Como se ha explicado líneas anteriores, en algunos casos se presenta un problema de asignación de privilegios a los usuarios que acceden a las aplicaciones, generado entre otras razones, por la ausencia de este concepto en los temarios de las diferentes asignaturas, sobre todo de bases de datos I y II. De la misma forma que se ha mencionado previamente, el problema se traslada hasta la implementación de las aplicaciones y su utilización por parte de los usuarios finales.

Desarrollo de la temática

Inicialmente es necesario conocer con mayor detalle algunos términos importantes de las bases de datos. El primer término que hay que conocer es el de sistema de base de datos, (Date,2001) lo define como un sistema cuya finalidad general es almacenar información y permitir a los usuarios recuperar y actualizar la información con base en peticiones. Un esquema bastante simple que puede ilustrar el entorno de una base de datos se muestra en la Figura 1. Existen distintos tipos de usuario en las bases de datos: usuarios normales, programadores de aplicaciones y usuarios especializados (Elmasri, 2007). Dependiendo de nuestro papel en el contexto de la aplicación actuaremos como un usuario determinado. Existe un tipo de usuario sobre el que recae la responsabilidad de definir el esquema inicial de la base de datos, restricciones, otorgar mantenimiento a la estructura y a la información. Otra tarea que tiene a cargo es la concesión de autorización para el acceso a los datos; es decir, determinar a qué partes de la base de datos puede acceder cada usuario. A este usuario se le conoce como el administrador de la base de datos o simplemente como raíz (root).

Un concepto importante es el Sistema Gestor de Bases de Datos (SGBD). Consiste en un conjunto de programas para acceder a datos, normalmente denominados Base de Datos y es relevante para una empresa o institución. Su objetivo principal es almacenar y recuperar información de manera eficiente (Silberschatz, 2006). Para este caso se utiliza el gestor de bases de datos MySQL.

Según Date (2001:10), una base de datos es un conjunto de datos persistentes que es utilizado por los sistemas de aplicación de alguna empresa dada.

En clases se le explica al alumno que, un sistema de base de datos se conforma de: usuarios, SGBD y una base de datos (esquema y datos).

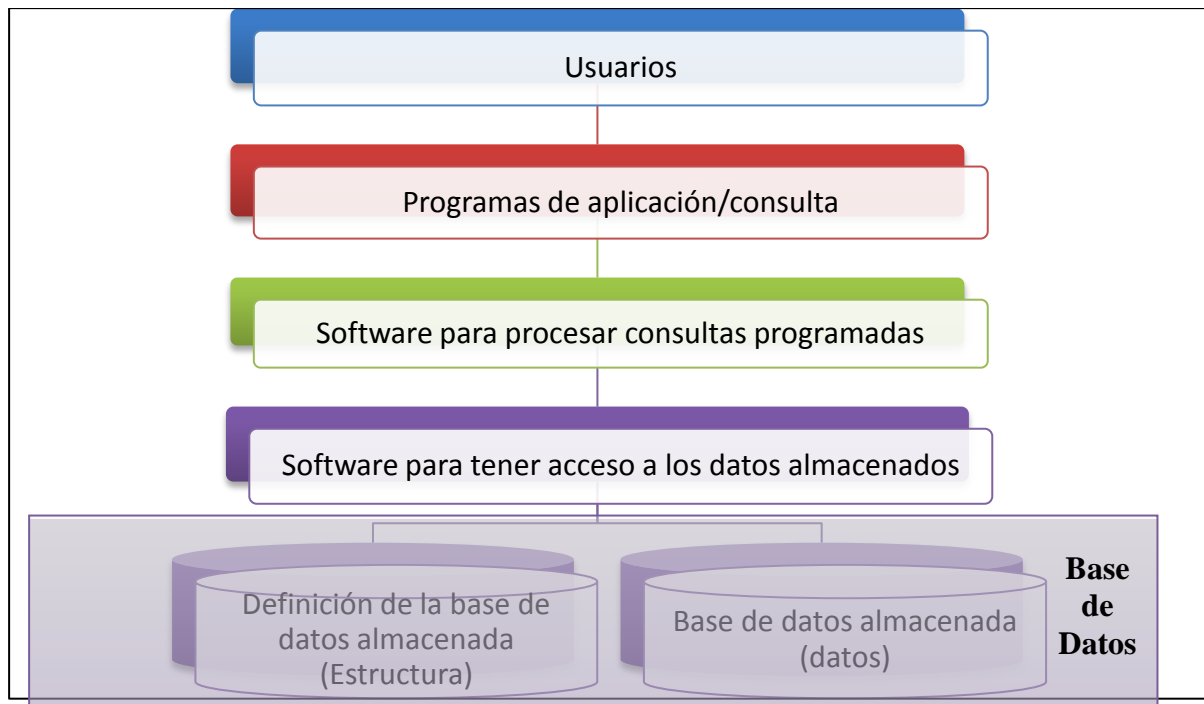


Figura 1 Sistema de Base de Datos

Cuando se desarrolla una aplicación, a cada usuario que la utiliza se le deben asignar derechos de acceso a la base de datos, lugar en donde se almacena la información. No puede tener los mismos privilegios que un usuario cuya única tarea puede ser la de consultar información, emitir un reporte, o enviar ciertos datos a impresión. Definir los privilegios repercute en algunos aspectos de seguridad en cuanto a que es muy importante que la información solo sea visualizada por los usuarios autorizados. Un erróneo diseño de la base de datos y de la aplicación de software puede generar problemas de este tipo.

Es de importancia mencionar que el lenguaje seleccionado para desarrollar la aplicación es PHP. Se define como un lenguaje de programación de uso general para el desarrollo de aplicaciones Web. Con todos estos elementos a la mano, se explicará una práctica realizada en el aula, en donde un grupo de alumnos (que a partir de este momento adoptan el papel de desarrolladores) es asignado a la creación de una aplicación que permita de una forma muy sencilla, consultar a los estudiantes de forma individual para saber las calificaciones que llevan en cada uno de sus cursos.

Cabe aclarar que al ser un proyecto de clase, de complejidad mediana o menos, la base de datos no es grande, y para el presente documento se toma como referencia solamente la parte necesaria para explicar nuestra temática.

Las actividades que realizan de manera paralela son las siguientes:

1. El diseño y desarrollo de la aplicación de software, que se conectará a la base de datos diseñada para mostrar a cada usuario la información solicitada.
2. El diseño de la base de datos que almacenará la información solicitada, de estudiantes, asignaturas cursadas, calificaciones obtenidas, etcétera.

Para el primer punto, los estudiantes se enfocaron en realizar actividades que se incluyen en el diseño y desarrollo de software, tales como:

- a) Análisis y especificación de requerimientos, en donde se especifican las funcionalidades que la aplicación cumplirá.
- b) Diseño del sistema, definiendo en un lenguaje de modelado la estructura global del sistema. Es decir, plasmar diagramas y modelos del funcionamiento del sistema, determinado en la etapa anterior.
- c) Implementación del sistema, que consiste en trasladar a un lenguaje de programación el diseño del sistema, se generan un conjunto de módulos o unidades de código que posteriormente serán integradas para tratarse como un sistema completo.
- d) Operación y mantenimiento, etapa en la que se pone en funcionamiento el sistema integrado en el paso anterior.

Referente al segundo punto, el diseño de la base de datos, se lleva a cabo un análisis detallado de la estructura de almacenamiento de la información. Dicha estructura debe seguir las reglas y lineamientos generales para el diseño en general de bases de datos. El resultado de este análisis es un diseño de la estructura que se muestra en la Figura 2.

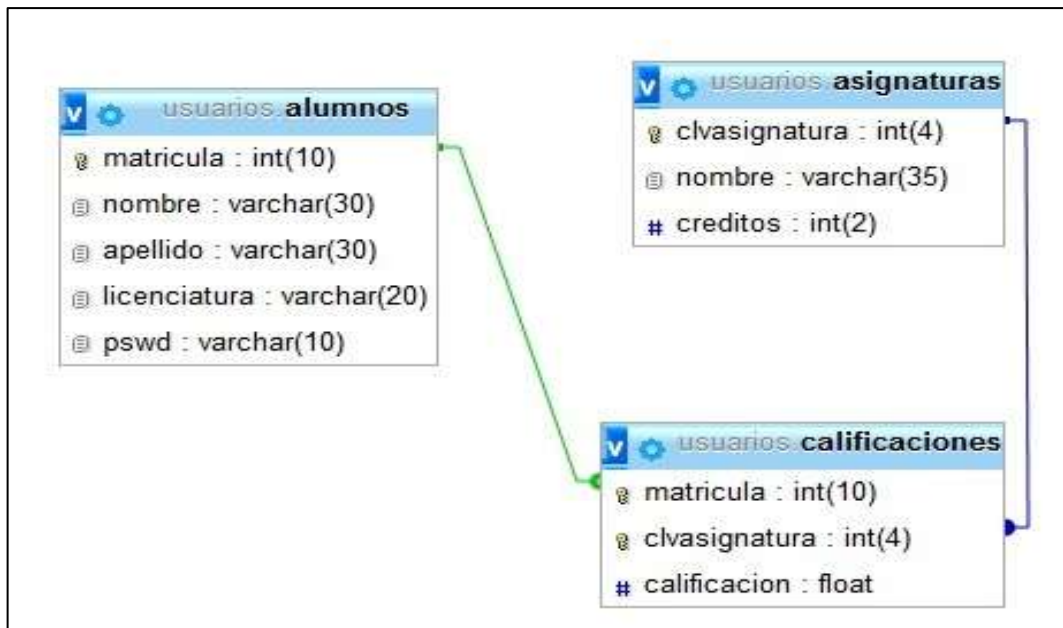


Figura 2 Fragmento de la Base de Datos

De acuerdo a lo observado, tenemos tres relaciones –alumnos, asignaturas y calificaciones– que almacenan la información a consultar para cada uno de los usuarios. Observamos que la relación alumnos tiene un atributo denominado pswd. A dicho atributo le corresponde una cadena de caracteres de longitud 10 que será utilizada como clave de acceso.

Centrando nuestra atención en esta relación, nos damos cuenta de que el atributo pswd servirá para que posteriormente en la aplicación que utilice la información, los usuarios “se conecten” y puedan visualizar la información.

matricula	nombre	apellido	licenciatura	pswd
2010060024	Keyla	Bustamante	Sistemas	keyla
2010060025	Neyma	Bautista	Sistemas	neyma
2010060026	María Martha	Canseco	Sistemas	maria
2010060027	Abigail	Cruz	Sistemas	abigail
2010060028	Margarita	Cruz	Sistemas	margarita
2010060029	Citlalli	Fabián	Sistemas	citlalli
2010060030	Mónica	Quiroz	Sistemas	monica
2010060031	Rigoberto	Bautista	Sistemas	rigoberto
2010060032	David	Pinacho	Sistemas	david
2010060033	Eliseo	Olivera	Sistemas	eliseo
2010060034	Aurora	Márquez	Sistemas	aurora
2010060035	Melissa	Martínez	Sistemas	melissa
2010060036	Alba	Domínguez	Sistemas	alba
2010060037	Carolina	Santos	Sistemas	carolina
2010060038	Karina	Jarquín	Sistemas	karina
2010060178	Antonio	Santiago Pazos	Sistemas	antonio
2010060185	Isabel	López	Sistemas	isabel
2010060193	Adriana	Cervantes	Sistemas	adriana
2010060201	Juan Carlos	Ventura	Sistemas	juancarlos
2010060205	Jesús	García	Sistemas	jesus
2010060213	Erika	Pérez	Sistemas	erika

La información que contiene esta tabla se muestra en la Figura 3.

El contenido de la información son datos de los estudiantes que ingresarán al sistema. Cada uno de los estudiantes debe tener una cuenta de usuario para solamente tener privilegios de acceso y consulta de la información. El problema –que hemos estado mencionando– radica en que indistintamente, por omisiones de diseño, codificación y un análisis detallado de la problemática, cada usuario accede como administrador, aunque en apariencia accede como un usuario con los derechos mínimos.

Por ejemplo, el usuario “Carolina Santos” visualiza la información que se muestra en la Figura 4 cuando consulta sus datos.

MATRICULA	NOMBRE	APELLIDO	ASIGNATURA	CALIFICACION
2010060037	Carolina	Santos	Pulsar para ordenar	7.2
2010060037	Carolina	Santos	Introducción a la Informática	8.6
2010060037	Carolina	Santos	Lógica Matemática	9.1
2010060037	Carolina	Santos	Historia del pensamiento filos	9.2
2010060037	Carolina	Santos	Diseño Estructurado de Algorit	8.9
2010060037	Carolina	Santos	Electrónica II	8.8
2010060037	Carolina	Santos	Matemáticas Discretas	9.1
2010060037	Carolina	Santos	Cálculo	9.1
2010060037	Carolina	Santos	Teoría General de Sistemas	9.1
2010060037	Carolina	Santos	Programación Estructurada	9.5
2010060037	Carolina	Santos	Arquitectura de Computadoras	9.7
2010060037	Carolina	Santos	Administración	8.4
2010060037	Carolina	Santos	Programación de Sistemas	9.6
2010060037	Carolina	Santos	Álgebra lineal	9.3
2010060037	Carolina	Santos	Estructuras de Datos	9.2

Figura 3 Registros del usuario "Carolina Santos"

Aparentemente no existe problema alguno. La información que visualiza es la información que le corresponde. Si esto es cierto, ¿cuál sería el problema entonces? La respuesta es simple, pero no implementarla en el software desarrollado sería de graves consecuencias.

Los alumnos, en su papel de desarrolladores de software, durante la etapa de codificación programan un módulo de conexión a la base de datos. La sintaxis de una de las instrucciones de conexión al gestor de bases de datos es como sigue:

```
<?php  
  
    mysql_connect("host","usuario","clave");  
  
?>
```

La función `mysql_connect` lo que hace es conectar a la aplicación de software desarrollada por los estudiantes con el gestor de bases de datos, lugar donde reside la información. Los parámetros que recibe la función para su correcta ejecución son:

1. Host es la dirección IP de la computadora en la que se encuentra instalado el gestor de bases de datos, en este caso MySQL. En el caso que estemos utilizando el gestor de manera local, el valor de este parámetro será `localhost`.
2. Usuario es el nombre de un usuario del gestor MySQL que tiene autorización para ejecutar ciertas tareas en la base de datos. El usuario administrador se conoce como `root` y tiene la facultad de realizar cualquier operación sobre los datos.
3. Clave es la contraseña que corresponde al usuario con el que se pretende tener acceso a la información.

Tomando los datos reales del problema, la instrucción de conexión quedaría así:

```
<?php  
  
    mysql_connect("localhost","root","cRoot");  
  
?>
```

Observando la instrucción, nos damos cuenta que en realidad se hace la conexión como un usuario `root` (usuario administrador de base de datos). El error está en que una vez conectado al gestor, los alumnos implementan un módulo de código en el cual para mostrar información se requiere "conectarse" como un "usuario final".

Por ejemplo, al ejecutar la aplicación, el usuario Carolina Santos se conecta como un administrador porque dentro del código se ejecuta la instrucción

```
mysql_connect("localhost","root","cRoot");
```

que le permite todos los derechos sobre el contenido del gestor de bases de datos, incluso el borrado de la propia base de datos y de otras contenidas en el servidor.

Una vez conectados como administrador (root), existe otra parte de código que utiliza los datos específicos del usuario Carolina Santos: matrícula y pswd (que corresponde al password). Las instrucciones se muestran a la continuación:

```
<?php
```

```
include_once "conexion.php";
```

```
?>
```

```
<form action="" method="post" class="registro">
```

```
    <div><label>Usuario:</label>
```

```
    <input type="text" name="usuario"></div>
```

```
    <div><label>Clave:</label>
```

```
    <input type="password" name="password"></div>
```

```
    <div><input type="submit" name="enviar" value="Ingresar"></div>
```

```
</form>
```

```
<?php
```

El código anterior nos da como resultado una interfaz como la que se muestra en la Figura 5 y que solicita el acceso al sistema.



The image shows a simple login form within a rectangular border. It contains two text input fields. The first field is labeled 'Usuario:' and contains the text 'Carolina'. The second field is labeled 'Password:' and contains six asterisks '*****'. Below these fields is a button labeled 'Enviar'.

Figura 4 Interfaz de acceso

En la Figura 5, se observa que para “conectarse” al sistema y a la base de datos, el usuario Carolina Santos teclea su nombre de usuario además del password que le corresponde. Esta clave es la que contiene como información en la relación alumnos, concretamente en el atributo o campo pswd.

De acuerdo con el ejemplo visto, cuando se ejecuta la aplicación, se tiene la idea de que se hace una conexión como un usuario normal. Sin embargo, de primera instancia el “usuario normal” se conecta como un usuario administrador (root).

El problema radica en que la misma situación de repite con cada uno de los usuarios que acceden al sistema y a la base de datos. Por lo tanto, para cada acceso se tienen todos los privilegios de la base de datos. Si uno de estos usuarios tuviera los conocimientos técnicos necesarios, puede en un momento determinado acceder a toda la información almacenada, no a la que le corresponde, y manipularla a sus intereses.

Solución planteada

Para evitar la problemática planteada debemos centrarnos con detalle en la enseñanza de este concepto. La confusión se elimina utilizando conceptos del lenguaje SQL y asignando privilegios a los distintos usuarios, diferentes a los del usuario administrador.

Como afirma Heurtel (2009:124) “para gestionar a los usuarios y los privilegios, son necesarios privilegios globales precisos (CREATE USER, GRANT OPTION, etcétera). De manera predeterminada, estos privilegios son otorgados a la cuenta root, puesto que esta última tiene todos los privilegios. En adelante, daremos por supuesto que la gestión de los usuarios y de los privilegios se efectúa mediante la cuenta root y no precisaremos qué privilegio se requiera para efectuar cada acción”.

El habilitado para gestionar a los usuarios nuevos es el propio usuario administrador. En cuanto a la sentencia GRANT, según lo expresa López (2000:14) “concede privilegios (autorizaciones y permisos), mientras que la sentencia REVOKE permite revocarlos”.

En versiones más recientes del gestor MySQL se utiliza CREATE USER para crear de manera explícita a un usuario. La sintaxis de la instrucción es la siguiente:

```
CREATE USER especificación_usuario [...]
```

```
especificación_usuario = nombre_usuario[@nombre_host] [IDENTIFIED BY [PASSWORD]  
'contraseña']
```

De acuerdo con Heurtel (2009:125), “nombre_usuario y contraseña son respectivamente el nombre y la contraseña de la nueva cuenta. Si la cláusula IDENTIFIED BY está ausente, la cuenta es creada sin contraseña”.

La anterior sintaxis es la que se debe ejecutar por cada usuario al que se desea otorgar acceso y posteriormente delimitar privilegios. Si consideramos al usuario “Carolina Santos”, la instrucción a ejecutar sería la siguiente:

```
CREATE USER 'Carolina'@'localhost'
```

```
IDENTIFIED BY 'carolina';
```

Una vez cumplido lo anterior, se ha dado de alta un nuevo usuario en el servidor de bases de datos. Esto lo podemos verificar en la siguiente consulta de usuarios que existen en el servidor (Ver Figura 6).

Host	User	Password	Select_priv	Insert_priv	Update_priv	Delete_priv	Create_priv	Drop_priv
localhost	root		Y	Y	Y	Y	Y	Y
127.0.0.1	root		Y	Y	Y	Y	Y	Y
:::1	root		Y	Y	Y	Y	Y	Y
localhost			N	N	N	N	N	N
%	Carolina	*B487C9B7701A91A6B44B2C910D3C980B29AC0220	N	N	N	N	N	N

Figura 5 Usuarios de la Base de Datos

Tenemos la información agregada al final de la tabla del usuario ‘Carolina’. Observemos con detalle que todavía no tiene privilegios. Solamente está agregado en la lista de usuarios del servidor. El paso siguiente es asignar los privilegios que le correspondan sobre una determinada base de datos, que puede ser uno de los siguientes:

- seleccionar información (SELECT),
- insertar información (INSERT),
- actualización de información (UPDATE),
- borrado de información (DELETE),
- creación de relaciones (CREATE) y,
- eliminación de relaciones (DROP).

Para el caso del usuario ‘Carolina Santos’ la instrucción sería:

```
GRANT SELECT ON `usuarios` . * TO 'Carolina'@'%';
```

En este momento, el usuario ‘Carolina’ tiene solamente privilegios de consulta sobre el contenido de la base de datos ‘usuarios’. Se ha erradicado el error inicial de conectar al usuario ‘Carolina’ como un usuario administrador (root).

Esta serie de acciones son las que los estudiantes (en su rol de desarrollador) deben definir en el ejemplo planteado. Con lo anterior, estamos obligando a que el usuario 'Carolina Santos' se conecte realmente con su propio nombre de usuario y contraseña. Así, la codificación de la conexión a la base de datos quedaría como sigue:

```
<?php  
  
    mysql_connect("localhost","Carolina","carolina");  
  
?>
```

De esta manera aseguramos que se conecte directamente el usuario 'Carolina Santos' y se eliminan algunos problemas de seguridad en el manejo de la información, ya que el usuario solo tendrá ciertos privilegios sobre la base de datos.

Para ampliar información sobre PHP y MySQL, se pueden consultar los manuales en línea (Achour, 2013) y (Oracle, 2013) respectivamente.

Conclusiones

El estudio y comprensión del diseño de bases de datos es fundamental en la formación de futuros licenciados en informática. Es importante que el profesor desarrolle las habilidades necesarias para la facilitar el conocimiento a sus estudiantes de manera que pueda aplicar los conocimientos adquiridos en la solución de problemas.

Un deficiente diseño en la base de datos puede provocar más problemas que los que se intentaban solucionar en un principio. En la licenciatura en informática de la Universidad de la Sierra Sur, hemos detectado un error común entre los estudiantes, el cual consiste en que tienen una perspectiva errónea sobre el término "usuario" en las bases de datos. El estudiante concibe al usuario como parte del esquema de la base de datos y no como un elemento de este, es decir, considera que el usuario es una tabla más de la base de datos, cuando la forma correcta es concebirlo como una persona que tendrá acceso a la base de datos (estructura y datos) mediante una conexión con ella, para lo cual tiene que indicar: la dirección del servidor de base

de datos, un nombre de usuario y una contraseña, que le permitirá acceder solo a ciertos recursos.

Hay que dejarle claro al estudiante que si los “usuarios” quedan registrados en una tabla, para tener acceso a ella se necesita una conexión a la base de datos, que a su vez requiere de un usuario.

La solución que se plantea es que desde un principio el estudiante defina a los usuarios como personas que utilizarán una base de datos mediante una conexión, y no como registros de una tabla, además de utilizar conceptos del lenguaje SQL para creación de usuarios, asignación y revocación de permisos sobre la base de datos.

Bibliografía

Date, C. J. (2001). *Introducción a los sistemas de Bases de Datos*. 7ª edición. Mexico: Pearson Education.

Silberschatz, Abraham. Korth, Henry y Sudarshan, S. (2006). *Fundamentos de Bases de Datos*. España: Mc.Graw Hill.

López, O. P., y Pons, P. B. (2000). *Gestión de Bases de Datos*. Valencia: Servicio de Publicaciones.

Elmasri, R., y Navathe, S. B. (2007). *Sistemas de Bases de Datos*. (2ª ed.). Madrid: Pearson Addison Wesley.

Heurtel, O. (2009). PHP y MySQL. *Domine el desarrollo de un sitio Web dinámico*. Barcelona: Ediciones ENI.

Beati, Hernan. (2011). PHP: creación de páginas web dinámicas(1ª ed.). Buenos Aires: Alfaomega.

Referencias electrónicas

Mehdi, Achour (2013). Manual de PHP . 18 de Diciembre de 2013, de The PHP Group Sitio web:
<http://www.php.net/manual/es/>

Oracle. (2013). MySQL 5.6 Reference Manual. 18 de Diciembre de 2013, de Oracle Sitio web:
<http://dev.mysql.com/doc/refman/5.6/en/index.html>