

Aplicación de una Base de Datos para el Taller de Ingeniería

LÓPEZ VÁSQUEZ ADRIÁN, GARCÍA MARTÍNEZ MITCHELL ALBERTO

Resumen— En este trabajo se presenta la aplicación de una base de datos para el taller de Ingeniería, con el fin de controlar la información que se maneja en dicho taller, ya sea sobre el material que se encuentra dentro de los almacenes, los horarios disponibles en los que podemos disponer de un laboratorio o el estado de las máquinas, así mismo, la capacidad de manejar la información proporcionada por los alumnos y profesores para tener un control sobre los préstamos y adeudos de los mismos.

I. INTRODUCCIÓN

En los talleres y laboratorios de la Facultad de Ingeniería de la Universidad La Salle, se cuenta con diferentes cantidades de equipos y materiales para que se puedan llevar a cabo prácticas de electrónica y manufactura. Al inicio de semestre los profesores pueden realizar pedidos para solicitar los materiales que requerirán los alumnos a lo largo del semestre. Al mismo tiempo los alumnos pueden solicitar equipo, material y maquinaria para poder desarrollar sus prácticas. En este artículo se revisará la aplicación que se utilizará para poder registrar los pedidos que se realizan por parte de los profesores y alumnos. Al mismo tiempo se revisará el diseño de la base de datos y las diferentes implementaciones que se utilizaron.

II. BASE DE DATOS

Se tomó como referencia una base de datos que crearon nuestros compañeros de séptimo semestre. Dicha base de datos y la aplicación fueron ampliadas para poder realizar pedidos y diferentes asignaciones al taller completo. El objetivo que se presenta en este trabajo es el desarrollar una ampliación de la base de datos anteriormente mencionada, con el fin de tener un control y registro de las actividades que se realizan todos los días en talleres. El taller cuenta con diferentes zonas de trabajo de los cuales el material y herramienta que se utiliza, es completamente diferente al de otras zonas. Como se llegan a presentar adeudos, una de las aportaciones de éste trabajo es la creación de tablas que son capaces de registrar la información del adeudo y del material a deber.

LÓPEZ VÁSQUEZ ADRIÁN, GARCÍA MARTÍNEZ MITCHELL ALBERTO pertenecen a la carrera INGENIERIA CIBERNETICA Y SISTEMAS COMPUTACIONALES de la Facultad de Ingeniería y realizaron el proyecto dentro del curso(s) BASE DE DATOS (Email: jd.silva.galindo@gmail.com).

El proyecto fue asesorado por JUAN DAMIAN SILVA y ZIZILIA ZAMUDIO BELTRÁN.

Los autores agradecen a La Facultad de Ingeniería por el apoyo brindado a la resolución de dudas.

III. CREACIÓN DE MODELO LÓGICOS Y FÍSICOS

El diseño del modelo lógico y físico, nos permite tener una representación de los elementos a utilizar en la base de datos. La diferencia de estos dos modelos, es que el modelo lógico representa los campos y tablas de manera general. Mientras que el modelo físico representa las tablas de manera normalizada. El normalizar las tablas nos permite tener un mejor orden de información y evita la redundancia de datos. Para ampliarlo observamos las diferentes tablas y campos que se utilizan en la Facultad de Ingeniería. La tabla más importante es préstamo, debido a que es la que se encarga de poder insertar registros de los pedidos que realizan los alumnos. Esta tabla tiene diferentes hijos porque el préstamo puede ser de algún material o equipo que se requiera. Al mismo tiempo solicita los datos del alumno que requiere el préstamo. La tabla préstamo tiene una tabla padre llamada almacén, para poder verificar la disponibilidad del material y/o equipo a prestar. En la figura 1 se puede observar el modelo lógico para la creación del sistema.

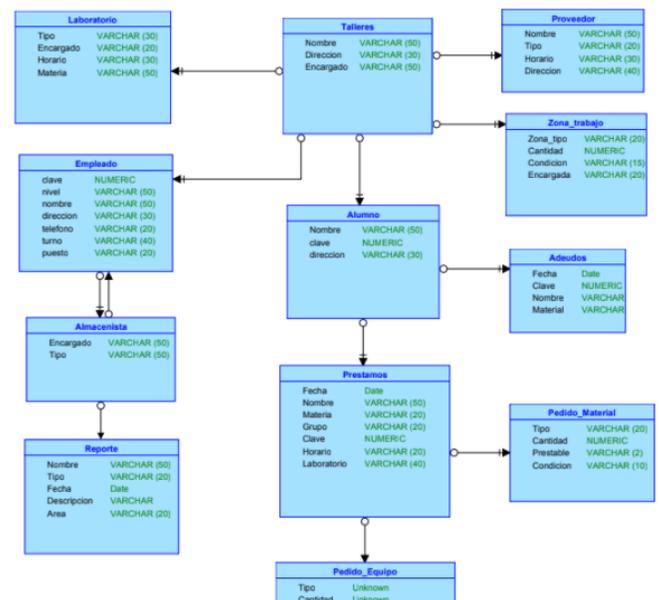


Figura 1. Modelo Lógico

IV. CREACIÓN DE TABLAS Y TRIGGERS.

La base de datos cuenta con 91 tablas, de las cuales 45 fueron tablas normales, 45 históricas y una tabla status. Las tablas normales contiene todos los campos normalizados y están relacionadas con otras tablas. Las tablas históricas se

utilizan para la parte de auditoría, en la cual se guardan los registros de los cambios que se realizan en las tablas. La tabla status se relaciona con cada una de las tablas normales, con el fin de tener un control de todas las tablas. Tiene una función de 'ACTIVO' o 'BORRADO' para que el objeto no se muestre al usuario en su momento.

Se implementaron triggers para cada tabla con el fin de actualizar e insertar información. Los disparadores se usaban antes de realizar la acción (Before Insert/Update) y se usaban cada que se realiza la acción (for each row). También se utilizó para la inserción en claves primarias, para realizar esto se creó una secuencia que generaba números consecutivos que eran disparados por un trigger cada que se insertaba un nuevo registro en cada tabla.

Para los triggers de inserción en tablas, se realizaron validaciones para checar si el registro ya existía, para mandar un mensaje de error. En los triggers de update se actualiza el dato que estaba antes por el nuevo registro. Cada trigger realiza inserción en la tabla histórica de la información que se registra, de mismo modo muestra quien y cuando modificó la información, así como la acción que se realiza.

V. SEGURIDAD

Se asignaron diferentes permisos a cada usuario para el uso de la aplicación. Se utilizaron los usuarios que se nos proporcionó al inicio de semestre para ingresar al servidor. Cada usuario tuvo un rol dependiendo los permisos que se le otorgaron.

VI. ASIGNACIÓN DE PERMISOS

Se creó una matriz de usuarios para poder determinar los nombres de usuarios, logins y el departamento al que pertenecen. En la tabla 1 se puede observar la tabla resultante.

USUARIO	LOGIN	DEPTO.
Marisol	1501	SISTEMAS
Isaac	1522	ENCARGADO
Mitchel	1500	ALMACÉN

Tabla 1. Matriz de usuarios

Los roles empleados que se le asignaron a cada usuario fueron los siguientes:

-Sistemas: Capacidad de conexión, crear objetos, consultar objetos, hacer respaldos, dar de baja y levantar la Base de Datos.

- Encargado: Capacidad de conexión, consultar objetos, dar de baja y levantar la Base de Datos.

- Almacén: Consultar objetos.

Los permisos asignados se realizaron por medio de un comando llamado GRANT, el cual asigna los diferentes permisos a los usuarios.

VII. ESPACIO FÍSICO DE MEMORIA

Se hace el cálculo en base a los tipos de datos almacenados, se suman por cada tabla existente, al mismo tiempo se incluyen tablas históricas (auditoría), llaves e índices. La suma de la tabla se multiplica por el número de tuplas (columnas) a insertar, después se hace una suma total de cada tabla incluyendo histórica y el status. En la siguiente tabla se puede observar el valor de cada tipo de dato

VARCHAR	1 BYTE X CARÁCTER
INT	4 BYTES
TIMESTAMP0	12 BYTES

Tabla 2. Tipo de dato.

A esa cantidad sin importar si es KB, MB, GB. Se le suma el peso del sistema operativo (10 GB) + dml (424 MB) +dbms(4.14 GB) + de la suma total de las tablas (30%). El 30% indica el crecimiento que se realiza de uno a dos años. Para el diseño de esta base de datos, su espacio físico es de 15 GB.

VIII. AUDITORÍA

La auditoría es el proceso de evaluar de manera independiente a un sistema o proceso de acuerdo a una metodología para emitir una opinión independiente. En este proyecto las tablas que se auditaron fueron taller, préstamo, alumno y pedido. Y se activaron por medio de las siguientes sentencias:

```
AUDIT SELECT ON TALLER WHENEVER
SUCCESSFUL;
audit insert on taller by access;
AUDIT SELECT ON PRESTAMO WHENEVER
SUCCESSFUL;
audit insert on prestamo by access;
AUDIT SELECT ON ALUMNO WHENEVER
SUCCESSFUL;
audit insert on alumno by access;
AUDIT SELECT ON PEDIDO WHENEVER
SUCCESSFUL;
audit insert on pedido by access;
```

IX. APLICACIÓN

La aplicación fue utilizada en un graficador llamado NetBeans, el cual ofrece herramientas para insertar frames(pantallas) y al mismo tiempo genera código. Ya que

solo se tiene que arrastrar lo tipos de objeto que necesita y programar los eventos de cada botón.

La aplicación cuenta con una pantalla principal que cuenta con un usuario y un login (figura 2).



Figura 2. Acceso a la aplicación

Al acceder al sistema, se detecta si el tipo de usuario es un profesor, alumno o empleado. Cada uno tiene una interfaz diferente y tiene una opción para poder realizar la acción deseada. Por ejemplo, en empleado (figura 3) si quiere realizar un reporte accederá a otro frame.



Figura 3. Frame de empleado

En el caso de realizar un reporte, aparecerá un frame solicitando algunos datos y sobretodo una sección para realizar el reporte requerido (Figura 4).



FIGURA 4. Frame de reporte

Si un alumno desea solicitar un préstamo. Se hará con base a la información que se solicite en la hoja de préstamo de los talleres de la Universidad La Salle. Esto se puede observar en la figura 5.



Figura 5. Frame de préstamo.

X. CONEXIÓN A LA BASE DATOS

El proyecto se realizó en NetBeans y para la conexión se necesitó un driver y las librerías de Oracle, el cual nos permite establecer la conexión con la base de datos. Además, esto permite que el programa pueda reconocer el lenguaje de Java. Para poder hacer la consulta de sentencia en la base de datos, se necesita un bloque try-catch. Donde necesita reconocer si se realizó la conexión, de lo contrario te mandará un error.

Primero se realiza la conexión con el driver por medio del puerto 1521. XE es la base de datos, el ejemplo_3 es nombre de la conexión a la que deseas acceder y al final su contraseña: `ConnectionmiConex=DriverManager.getConnection("jdbc:oracle:thin:@localhost:1521:XE","ejemplo_3","bd");`

Para crear un statement, que va a ser la variable con la que te podrás conectar con Oracle y hacer ejecuciones, se inserta el siguiente comando:

```
Statement miStatement = miConex.createStatement();
```

Por último, el 'resultset' será la variable con la que mandarás a llamar el 'Statement', donde irá la acción a realizar (consulta, inserción, actualización) y será guardado todo en esa variable: `ResultSet nombre = miStatement.executeQuery("SELECT * FROM PERSONA");`

XI. TRANSACCIONES

Cuando hablamos de transacción, estamos hablando de una operación de diverso tipo que se realiza entre dos o más partes y que supone el intercambio de bienes o servicios a cambio del capital correspondiente. En esta aplicación se utilizó el servicio de préstamo de material, al momento de tomar o surtir el material del almacén.

Donde si la operación fue correcta se realiza un 'commit' dentro de la aplicación, de lo contrario se realizará un 'rollback' y si se está insertando y consultando, será una por persona. Si no dará inconsistencia en la base de datos.

XII. CONCLUSIÓN

El objetivo de este proyecto se cumplió al ampliar una base de datos que fuera capaz de tener una cobertura completa en cuanto a los préstamos y zonas de trabajo presentes en el Taller de Ingeniería de la Universidad La Salle. Además, se realizó una interfaz completa de todas las tablas creadas para que profesores, alumnos y empleados del taller puedan hacer uso de ella.

XIII. AGRADECIMIENTOS

Agradecemos al Profesor Juan Damián Silva Galindo y Zizilia Zamudio Beltrán por las aportaciones, comentarios y sugerencias que realizaron a lo largo de la elaboración de este proyecto y los conocimientos que nos proporcionaron en el semestre.

Del mismo modo, agradecemos a Isaac Martínez Aldama, el jefe de turno matutino y Magda Karina López López y su equipo de proyecto, por proporcionarnos información del diseño de su base para utilizarlo como referencia.

XIV. REFERENCIAS

- [1] ABRUTSKY, (2012), "Bases de datos", AlfaOmega.
- [2] Rob, P. (2003), "Sistemas de bases de datos: diseño, implementación y administración", Paraninfo.
- [3] Silberschatz, A. (2002). "Fundamentos de Bases de Datos", Madrid: McGRAW-HILL.
- [4] David M. Kroenke (2202), "Procesamiento de Base de Datos, Fundamentos, Diseño e Implementación", Pearson Prentice Hall.
- [5] Ramez Elmasri, Shamkant B. Navathe (2007), "Fundamentos de Sistemas de bases de datos", Pearson Educación..