INFORME DE CONSTRUCCIÓN DE BASE DE DATOS

SISTEMA DE GESTIÓN DE BIBLIOTECAS

Carlos Rene Angarita

Base de Datos

**Hoja de Control de Actualizaciones del Documento**

|  |  |  |
| --- | --- | --- |
| **Versión** | **Fecha** | **Descripción de la modificación** |
| 01 | 26 de Octubre 2019 | Se inicia y finaliza el documento, especificando roles, y el paso a paso del proyecto |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **ELABORO** |  | **REVISO** |  | **APROBO** |
| Jhoan Alfredo Mancilla Blanco |  | Eduardo José Pico Claro |  | Cristian David López Guarín |
| Desarrollador 1 |  | Desarrollador 2 |  | Desarrollador 3 |

Contenido

[1. Definición del Sistema de Gestión de Base de Datos 4](#_Toc480145034)

[1.1. Selección del SGBD 4](#_Toc480145035)

[1.2. Instalación del SGBD 4](#_Toc480145036)

[2. Construcción del Sistema 4](#_Toc480145037)

[2.1. Creación de la base de datos 4](#_Toc480145038)

[2.2. Creación de tabla 4](#_Toc480145039)

[2.3. Ingreso de registros a la base de datos 5](#_Toc480145040)

[2.4. Copias de respaldo 5](#_Toc480145041)

# Definición del Sistema de Gestión de Base de Datos

## Selección del SGBD

Se seleccionó MySQL V 5.7 ya que es un sistema gestor de base de datos de uso libre, muy común, de fácil instalación y uso, además está bastante optimizado.

* MySQL fue creado por una compañía sueca llamada MySQL AB 1995
* Sun adquirió MySQL AB por mil millones de dólares en el año 2008
* Oracle compró Sun en 2010 y con ello adquirió MySQL
* En 2012, MySQL fue bifurcado en MariaDB por el fundador Michael Widenius bajo la compañía Monty Program Ab
* MariaDB reemplaza MySQL para la mayoría de las distribuciones en el año 2013
* Monty Program Ab se fusionó con SkySQL- 2013
* SkySQL Ab renombrado a MariaDB Corporation- 2014

**Características de MySQL**

* MySQL es un sistema DBMS impulsado por la comunidad.
* Compatible con varias plataformas que utilizan todos los idiomas principales y middleware
* Ofrece soporte para control de concurrencia de versiones múltiples
* Cumple con el estándar ANSI SQL
* Permite la replicación SSL basada en registros y en disparadores
* Orientado a objetos y compatible con ANSI-SQL2008
* Diseño multicapa con módulos independientes.
* Completamente multiproceso, utilizando hilos de kernel
* Servidor disponible en DB incorporado o modelo de servidor cliente
* Ofrece herramientas integradas para análisis de consultas y análisis de espacio.
* Puede manejar cualquier cantidad de datos, hasta 50 millones de filas o más.
* MySQL se ejecuta en muchas variedades de UNIX, así como en otros sistemas que no son UNIX como Windows y OS / 2.

**¿Por qué usar MySQL?**

* Soporta características como replicación maestro-esclavo, escalamiento horizontal
* Admite informes de descarga, distribución de datos geográficos, etc.
* Muy baja sobrecarga con el motor de almacenamiento MyISAM cuando se usa para aplicaciones de lectura mayoritaria
* Soporte para el motor de almacenamiento de memoria para tablas de uso frecuente.
* Caché de consultas para declaraciones utilizadas repetidamente
* Puede aprender y solucionar problemas de MySQL fácilmente desde diferentes fuentes, como blogs, libros blancos y libros.

**Desventajas de usar MySQL**

* Las transacciones relacionadas con el catálogo del sistema no cumplen con ACID
* En algún momento Un bloqueo del servidor puede dañar el catálogo del sistema
* Sin módulo de autenticación conectable que impide la cuenta administrada centralmente
* No hay soporte para roles, por lo que es difícil mantener los privilegios para muchos usuarios
* Los procedimientos almacenados no se pueden almacenar en caché
* Las tablas utilizadas para el procedimiento o disparador siempre están prebloqueadas.

Referencia web: <https://www.guru99.com/postgresql-vs-mysql-difference.html>

## Instalación del SGBD

El proceso de instalación de MySQL en nuestro proyecto, es en un servidor EC2 de Amazon con Sistema Operativo Ubuntu 16, al conectarse mediante conexión SSH al servidor, a través de consola podemos descargar, instalar y configurar el SGBD con las siguientes líneas:

(Previamente el servidor, debe tener instalado Apache 2 y PHP)

* apt-get install mysql-server (Acá se digita la clave que se usará para la posterior conexión con el gestor)
* apt-get install phpmyadmin (Para el entorno gráfico)
* nano /etc/apache2/apache2.conf (Para configurar el Apache y funcione correctamente)
* Include /etc/phpmyadmin/apache.conf (Acá se le indica al Apache, la ruta de phpmyadmin)
* sudo service apache2 restart (Usamos sudo para darle privilegios root al comando, y con ello se reinicia el servidor Apache)

Ahora para crear un Usuaro DBA nos conectamos por consola al servidor de MySQL así:

* mysql -u root –p (Pedirá la clave que previamente se estableció)
* CREATE DATABASE latinscorts; (Creamos la base de datos)
* CREATE USER superadmin@localhost IDENTIFIED BY 'password'; (Se crea el usuario original)
* GRANT ALL PRIVILEGES ON latinscorts.\* TO wordpressuser@localhost;FLUSH PRIVILEGES; (Le damos al usuario “superadmin” todos los privilegios sobre la base de datos “latinscorts”)
* FLUSH PRIVILEGES; (Refrescamos los privilegios)
* Exit (Salimos del servidor MySQL)

Para acceder de manera fácil a phpmyadmin, es necesario poner en la url del navegador “/phpmyadmin”, el usuario por default es “root” y la contraseña es la misma que se ha designado anteriormente, una vez dentro, el usuario puede crear usando phpmyadmin nuevas bases de datos, tablas, columnas, etc… además de contar con una vista llamada “Diseñador” la cual brinda un preview al ORM.

# Construcción del Sistema

## Creación de la base de datos

En los siguientes puntos se enseñará el DDL para la creación de la BD y las relaciones.

## Creación de tabla

A continuación se muestra lista de tablas creadas que forman parte de este proyecto. El detalle de cada una de ellas puede verse en el apartado de diseño o bien en el propio script de creación de tablas.

Pais,

Biblioteca(\*), Empleado(\*), TBLOG(\*), Coleccion(\*), Material(\*), Recurso(\*), Autor(\*), RecursoAutor, Ejemplar(\*), Tipo(\*), Socio(\*), Estado(\*), HistEstado, Prestamo, Reserva, Limite, Transaccion(\*), Pago(\*).

Las tablas marcadas con (\*) requieren de la creación de una secuencia por ser de clave autonumérica.

A continuación veremos la definición de un par de tablas contenidas dentro del script. En ellas se puede ver cómo se establece la integridad referencial tal como se definió en el apartado de diseño lógico.

|  |
| --- |
| CREATE TABLE `usuario` (  `id\_usuario` int NOT NULL AUTO\_INCREMENT,  `nickname` varchar(150) NOT NULL,  `correo` varchar(150) NOT NULL,  `fecha\_registro` datetime NOT NULL ON UPDATE CURRENT\_TIMESTAMP,  `id\_ciudad` int NULL,  `tipo` varchar(100) NOT NULL,  `ip\_usuario` varchar(250) NULL,  PRIMARY KEY (`id\_usuario`)  ); |

|  |
| --- |
| CREATE TABLE `anunciante` (  `id\_usuario\_anunciante` int NOT NULL,  `telefono` numeric(10,0) NOT NULL,  `edad` int(4) NULL,  `descriptcion\_perfil` longtext NULL COMMENT 'descripcion del perfil',  `foto\_perfil` varchar(250) CHARACTER SET utf8 NULL COMMENT 'Esta sera la foto que tomaremos como perfil',  `sexo` int NULL,  `verificado` binary(1) NULL DEFAULT 0,  `valoracion` double(5,2) NULL,  PRIMARY KEY (`id\_usuario\_anunciante`)  ); |

## Ingreso de registros a la base de datos

(Sentencia que agrega un usuario)

INSERT INTO `usuario`(`id\_usuario`, `nickname`, `correo`, `fecha\_registro`, `id\_ciudad`, `tipo`,`ip\_usuario`)VALUES('01','admin','admin@latinscorts.com','15-10-2018','10','moderador','');

(Sentencia que agrega un usuario moderador)

INSERT INTO `moderador`(`id\_moderador\_usuario`, `nombre`, `nivel`, `telefono`, `documento`,`direccion`) VALUES ('01','Jhoan Alfredo Mancilla Blanco','1','3112037610','1093787904','Cll7 Dn #15 e -21');