***Diagrama entidad relacion***

***Codigo del mismo***

DROP TABLE IF EXISTS [Catálogo colaboradores];

DROP TABLE IF EXISTS [Catálogo de puestos];

DROP TABLE IF EXISTS [Catálogo de área ];

CREATE TABLE [Catálogo colaboradores] (

[Nombre completo] VARCHAR (100) NOT NULL,

[ID colaborador] INT NOT NULL,

[Puesto] VARCHAR NOT NULL,

[Area] INT NOT NULL,

[Nivel jerarquico] VARCHAR (50) NOT NULL,

[Fecha de registro] DATETIME NOT NULL,

[Fecha de status] DATETIME NOT NULL,

[Fecha de último ingreso] DATETIME NOT NULL,

[Estatus (activo/inactivo)] VARCHAR (50) NOT NULL);

CREATE TABLE [Catálogo de puestos] (

[ID puesto] INT NOT NULL,

[Descripción del puesto] VARCHAR (100) NOT NULL,

[Fecha de registro] DATETIME NOT NULL);

CREATE TABLE [Catálogo de área ] (

[ID área ] INT NOT NULL,

[Descripción del area] VARCHAR (200) NOT NULL,

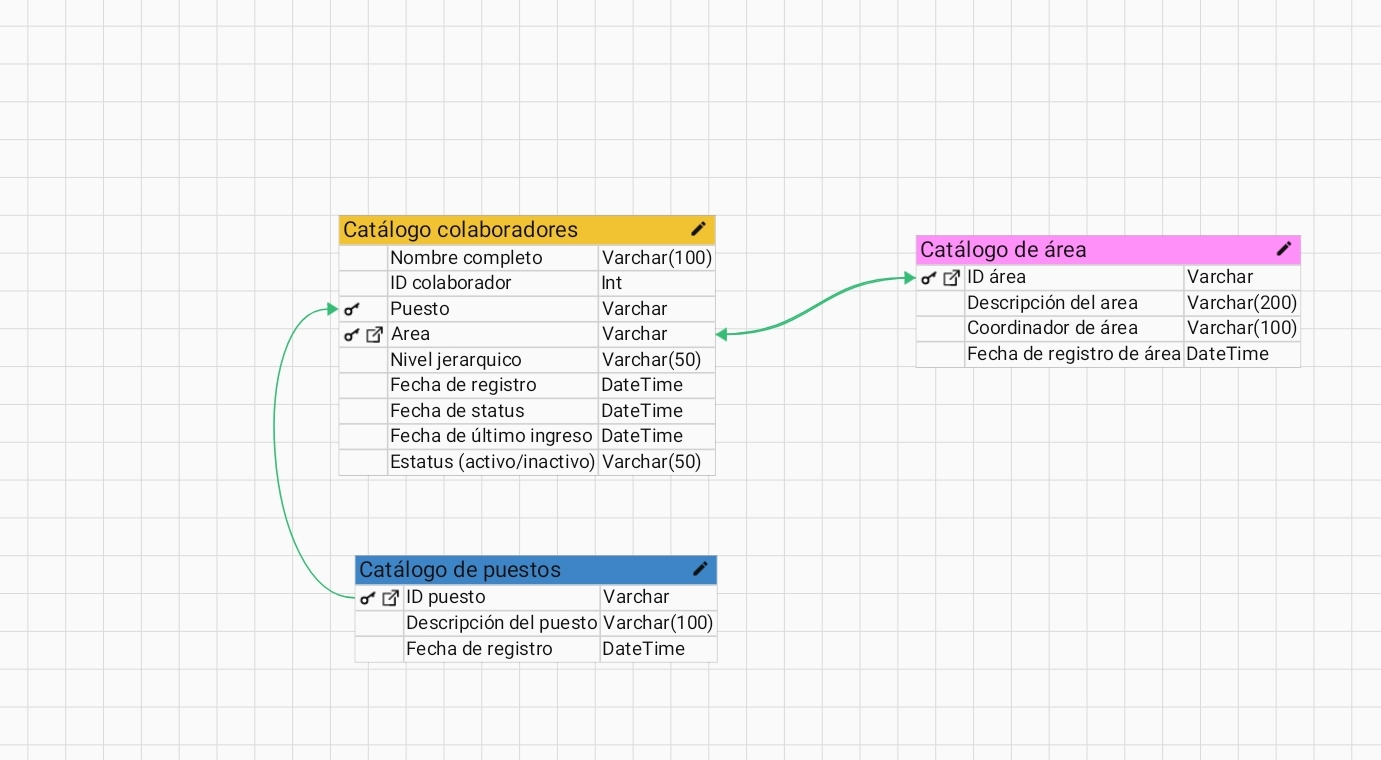
[Coordinador de área ] VARCHAR (100) NOT NULL,

[Fecha de registro de área ] DATETIME NOT NULL);

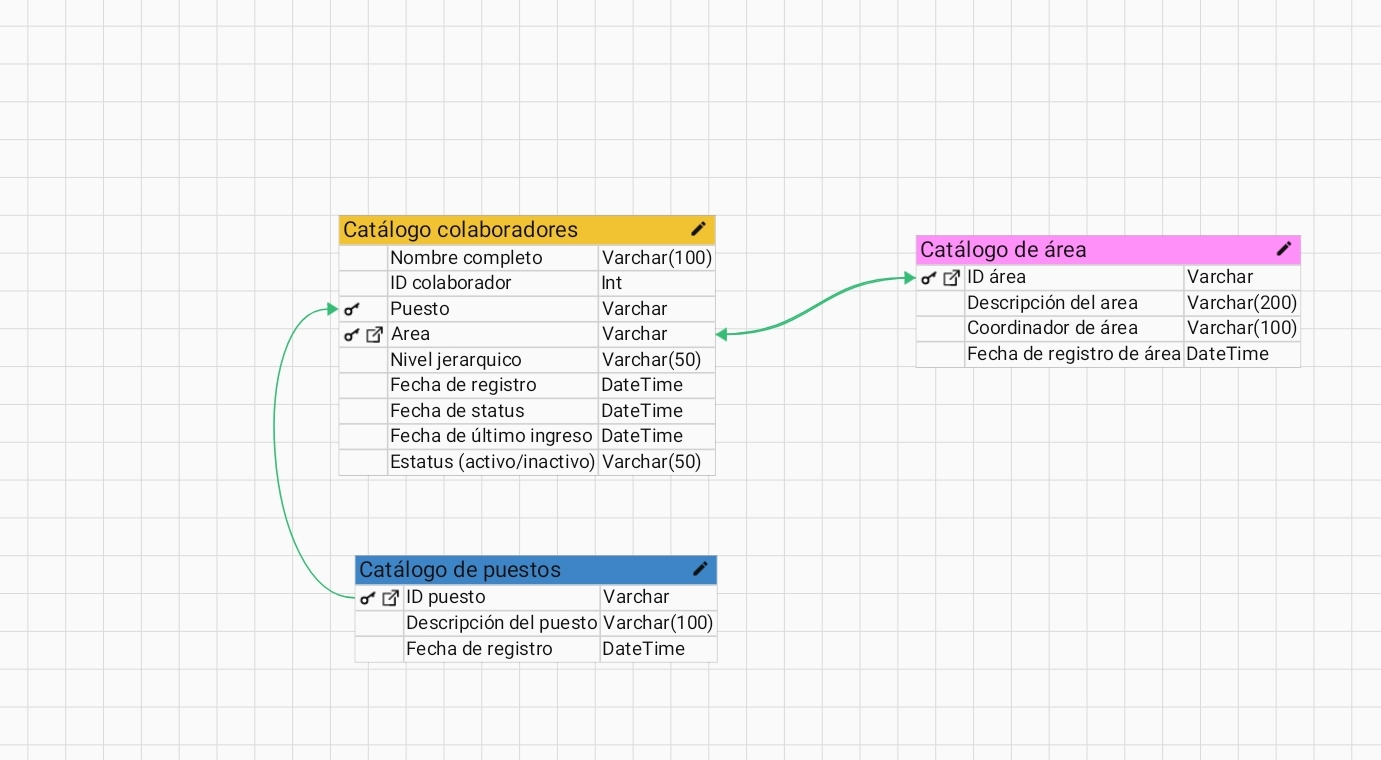
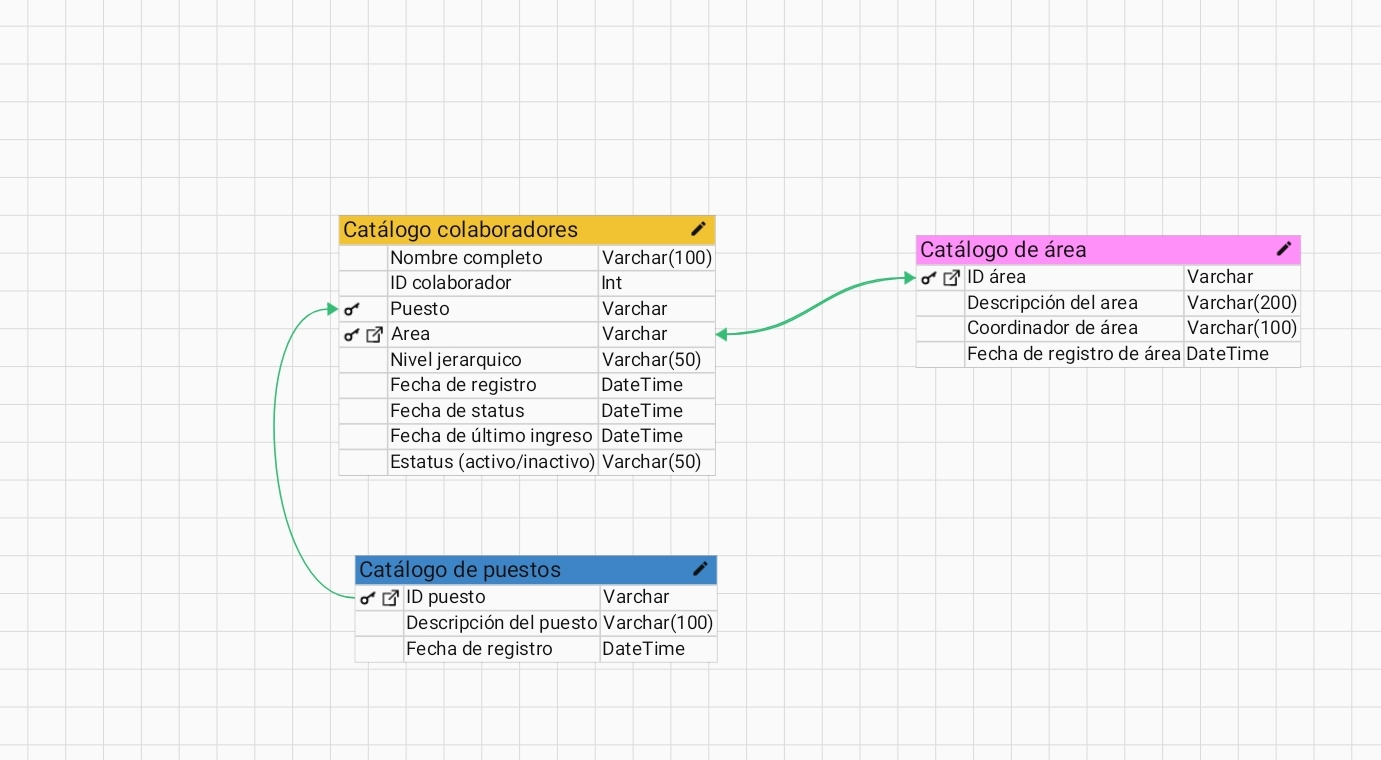
***Proceso de almacenado***

La manera correcta y mas eficiente para el proceso de almacenado es realizandolo en particiones, es decir por medio de uso de modulos o tablas para mayor eficiencia y seguridad para los datos.

Ejemplo:



Particion 1



Particion 2

Particion 3

Donde se busca realizar copias de seguridad por separado para evitar dañar documentos o datos que se encuentran en perfecto estado, de forma practica para que se entienda el consepto tienes en tu smarthphone muchas fotos donde borras accidentalmente una imagen y el proceso es bastante lento si deseas restaurar todo, en cambio sabes de que carpeta fue borrada y solo desde los ajustes de respaldo le das aplicar copia de seguridad y sera en menor tiempo la restauracion, pasa lo mismo en estas bases de datos pero con la diferencia que aqui son millones de datos los cuales debes tener en orden y si falla uno pueda hacer una falla en cadena que afecte la productividad de la empresa por ende peridas millonarias.

**Proceso de respaldo**

Ya que se prevee que esta base de datos se expanda y contenga los minimos errores posibles se implementara de forma fisica dentro de la misma empresa para asi contar con el control total de la informacion para crear un sistema que funcione en conjunto con la parte fisica que son las tiendas y la parte de recursos humanos (donde se recluntan nuevos colaboradores) un ejemplo para que sea entendible el mismo, cuando se realizan mantenimientos de empresas externas, o por un equipo de inventarios que funcionan en distintas tiendas en una zona en concreto se tienen que dar de alta o poner asistencia que estan activos en dicha tienda para que asi en caso de incidencias externas se tenga contemplado que estos individuos estaban dentro de dicha tienda.

¿Por que no hacerlo en la nube?

Como lo decia en el ejemplo anterior puede alguien externo registrarse para que tenga acceso al sistema y asi poder realizar modificaciones en el mismo sistema, si en el caso que un pirata informatico encuentre la forma de interceptar la central que se comunica con la nube y registrarse como usuario y se diriga a una sucursal y cometa fraude ya sea de articulos gratis o escalar a realizar depositos a su cuenta bancaria y perjudicar a la tienda.

¿Por cual irias tu?

***Codigo para crear respaldo***

function backDb($host, $user, $pass, $dbname, $tables = '\*'){

//make db connection

$conn = new mysqli($host, $user, $pass, $dbname);

if ($conn->connect\_error) {

die("La conexión falló: " . $conn->connect\_error);

}

//get all of the tables

if($tables == '\*'){

$tables = array();

$sql = "SHOW TABLES";

$query = $conn->query($sql);

while($row = $query->fetch\_row()){

$tables[] = $row[0];

}

}

else{

$tables = is\_array($tables) ? $tables : explode(',',$tables);

}

//getting table structures

$outsql = '';

foreach ($tables as $table) {

// Prepare SQLscript for creating table structure

$sql = "SHOW CREATE TABLE $table";

$query = $conn->query($sql);

$row = $query->fetch\_row();

$outsql .= "\n\n" . $row[1] . ";\n\n";

$sql = "SELECT \* FROM $table";

$query = $conn->query($sql);

$columnCount = $query->field\_count;

// Prepare SQLscript for dumping data for each table

for ($i = 0; $i < $columnCount; $i ++) {

while ($row = $query->fetch\_row()) {

$outsql .= "INSERT INTO $table VALUES(";

for ($j = 0; $j < $columnCount; $j ++) {

$row[$j] = $row[$j];

if (isset($row[$j])) {

$outsql .= '"' . $row[$j] . '"';

} else {

$outsql .= '""';

}

if ($j < ($columnCount - 1)) {

$outsql .= ',';

}

}

$outsql .= ");\n";

}

}

$outsql .= "\n";

}

// Save the SQL script to a backup file

$backup\_file\_name = $dbname . '\_backup.sql';

$fileHandler = fopen($backup\_file\_name, 'w+');

fwrite($fileHandler, $outsql);

fclose($fileHandler);

// Download the SQL backup file to the browser

header('Content-Description: File Transfer');

header('Content-Type: application/octet-stream');

header('Content-Disposition: attachment; filename=' . basename($backup\_file\_name));

header('Content-Transfer-Encoding: binary');

header('Expires: 0');

header('Cache-Control: must-revalidate');

header('Pragma: public');

header('Content-Length: ' . filesize($backup\_file\_name));

ob\_clean();

flush();

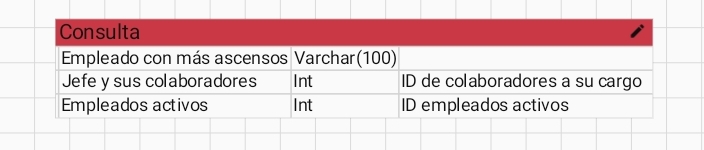
readfile($backup\_file\_name);

exec('rm ' . $backup\_file\_name);

}

?>

**Consultas**



Para faculitar la consulta de datos de colaboradores y evitar la saturacion de informacion en la pantalla se reduce a medio de los ID de cada colaboradory la opcion de ver la informacion desplegada de cada colaborador al hacer clic en ID