




SAIA

Informe de migración de datos

Versión: 0100-2022

	SAIA Informe de migración de datos	SENA
---	---	-------------

HOJA DE CONTROL

Organismo	SENA		
Proyecto	Sistema Autónomo de Información Automotriz		
Entregable	Informe de migración de datos		
Autor	SAIA		
Aprobado por		Fecha Aprobación	DD/MM/AAAA
		Nº Total de Páginas	23

REGISTRO DE CAMBIOS

Versión	Causa del Cambio	Responsable del Cambio	Fecha del Cambio
0100	Versión inicial	Sergio Leon, Brayan Chavarro y Sergio León	25/02/2023

CONTROL DE DISTRIBUCIÓN

Nombre y Apellidos
Sergio Fernando Limas
Sergio Andres Leon
Brayan Stiven Chavarro Morales



	<p align="center">SAIA</p> <p align="center">Informe de migración de datos</p>	<p align="center">SENA</p>
---	--	-----------------------------------

Tabla de contenido

1	INTRODUCCIÓN	4
1.1	Objeto.....	4
1.2	Alcance	4
2	Plan de instalación.....	5
2.1	Elección del sistema operativo.....	5
2.2	Instalarlo en Linux CentOS 7.	5
2.3	Instalación de nuestra base de datos en MariaDB.	9
3	Plan de respaldo	12
3.1	Dumps periodicos	12
4	Plan de migración.....	15
4.1	Migración a SQL server.....	15

	<p style="text-align: center;">SAIA</p> <p style="text-align: center;">Informe de migración de datos</p>	<p style="text-align: center;">SENA</p>
---	--	--


1 INTRODUCCIÓN

1.1 Objeto

El Sistema autónomo de información automotriz (SAIA) originalmente trabaja con el SGBD MySQL más concretamente con su derivado de código libre MariaDB, este será nuestro punto de partida para planear la migración de datos partiendo de que migrar datos se refiere a hacer nuestra base de datos completamente funcional en otro gestor diferente a MariaDB, para ello estableceremos el procedimiento para migrar nuestra base de datos a otro gestor planificando 3 escenarios planteados por el entregable macro al que hace referencia este documento los mismos están directamente relacionados e irán en el siguiente orden: Plan de instalación, Plan de respaldo y plan de migración, en ellos se identificara el plan para instalar nuestra base de datos en el gestor original, un plan de respaldo donde indicaremos una forma en la que se debería realizar la copia de seguridad de la base de datos y el plan de migración donde explicaremos la forma de migrar nuestra base de datos al SGBD SQL Server.

1.2 Alcance

Este documento va dirigido a los responsables de la gestión de infraestructura de servidores que más tarde alojaran a nuestro sistema de información; como una guía para el correcto despliegue de la aplicación probado por nosotros.

	<p style="text-align: center;">SAIA</p> <p style="text-align: center;">Informe de migración de datos</p>	<p style="text-align: center;">SENA</p>
---	--	--

2 Plan de instalación

2.1 Elección del sistema operativo

Para instalar nuestra base de datos debe elegir un sistema operativo, recomendamos LINUX ya que es un sistema operativo de código abierto perteneciente a la free software foundation así que es totalmente gratis y es el que mejor funciona con nuestro gestor pues además de ser muy versátil en la gestión de recursos suele asociarse los dos en la construcción de una pila LAMP para ejecutar aplicaciones con PHP y bases de datos relacionales de MySQL, puede utilizar distribuciones como CentOS, Debian o Ubuntu, o si lo desea RedHat pero para usarlo si deberá pagar por licencia.

Puede también instalarlo en sistemas operativos Windows como alternativa.

2.2 Instalarlo en Linux CentOS 7.

Nuestro sistema funciona de manera adecuada si se monta en una pila LAMP (Linux, Apache, MySQL y PHP), así que para este ejemplo guiaremos la instalación en un sistema operativo Linux conocido como CentOS, este sistema es totalmente free siendo la alternativa de código abierto del popular Linux empresarial RedHat es un sistema poderoso y seguro que brinda una excelente base para la instalación y ejecución de nuestra BD.

Para instalarlo en CentOS 7 siga este paso a paso

- inicialmente actualice los repositorios.

Utilice el comando **yum -y update** para actualizar repositorios


```
[root@localhost ~]# yum -y update
```

- Luego de actualizar los repositorios instale Mariadb

Utilice el comando **yum -y install mariadb mariadb-server**

```
[root@localhost ~]# yum -y install mariadb mariadb-server
```

Debe esperar a que se instalen los paquetes necesarios para el correcto funcionamiento

 <p>SISTEMA AUTÓNOMO DE INFORMACIÓN AUTOMOTRIZ</p>	<p>SAIA</p> <p>Informe de migración de datos</p>	<p>SENA</p>
---	--	-------------

de mariadb.

```

--> Paquete mariadb-server.x86_64 1:5.5.68-1.el7 debe ser instalado
--> Resolución de dependencias finalizada

Dependencias resueltas

=====
Package                Arquitectura  Versión      Repositorio  Tamaño
-----
Instalando:
mariadb                x86_64       1:5.5.68-1.el7  base         8.8 M
mariadb-server          x86_64       1:5.5.68-1.el7  base        11 M
=====

Resumen de la transacción:
=====
Instalar 2 Paquetes

Tamaño total de la descarga: 20 M
Tamaño instalado: 107 M
Downloading packages:
(1/2): mariadb-5.5.68-1.el7.x86_64.rpm | 0.0 MB 00:00:07
(2/2): mariadb-server-5.5.68-1.el7.x86_64.rpm | 11 MB 00:00:10
-----
Total                               1.9 MB/s | 20 MB 00:00:10

Running transaction check
Running transaction test
Transaction test succeeded
Running transaction:
  Instalando : 1:mariadb-5.5.68-1.el7.x86_64                               1/2
  Instalando : 1:mariadb-server-5.5.68-1.el7.x86_64                       2/2
  Comprando  : 1:mariadb-server-5.5.68-1.el7.x86_64                       1/2
  Comprando  : 1:mariadb-5.5.68-1.el7.x86_64                             2/2

Instalado:
mariadb.x86_64 1:5.5.68-1.el7          mariadb-server.x86_64 1:5.5.68-1.el7

¡Listo!
[root@localhost ~]#

```

- Acto siguiente inicie mariadb con el sistema operativo, para que cada que se reinicie el servidor el gestor de bases de datos inicie con el:

Utilice el comando **systemctl enable mariadb**

```

[root@localhost ~]# systemctl enable mariadb
Created symlink from /etc/systemd/system/multi-user.target.wants/mariadb.service to /usr/lib/systemd/system/mariadb.service.
[root@localhost ~]#

```

Se creará el enlace simbólico para iniciar el servicio con el servidor.

- Inicie el servicio de bases de datos


Utilice el comando **systemctl start mariadb** para iniciar el servicio

```

/systemd/mariadb.service.
[root@localhost ~]# systemctl start mariadb
[root@localhost ~]#

```

- Utilice el comando **systemctl status mariadb** para verificar el estado del servicio de

 <p>SISTEMA AUTÓNOMO DE INFORMACIÓN AUTOMOTRIZ</p>	<p>SAIA</p> <p>Informe de migración de datos</p>	<p>SENA</p>
---	--	-------------

MariaDB.

```
[root@localhost ~]# systemctl status mariadb
■ mariadb.service - MariaDB database server
   Loaded: loaded (/usr/lib/systemd/system/mariadb.service; enabled; vendor preset: disabled)
   Active: active (running) since jue 2023-03-16 21:12:09 -05; 5min ago
   Process: 2141 ExecStartPost=/usr/libexec/mariadb-wait-ready $MAINPID (code=exited, status=0/SUCCESS)
   Process: 2106 ExecStartPre=/usr/libexec/mariadb-prepare-db-dir %n (code=exited, status=0/SUCCESS)
   Main PID: 2140 (mysqld_safe)
   CGroup: /system.slice/mariadb.service
           └─2140 /bin/sh /usr/bin/mysqld_safe --basedir=/usr
             └─2305 /usr/libexec/mysqld --basedir=/usr --datadir=/var/lib/mysql --plugin-dir=/usr/lib...

mar 16 21:12:07 localhost.localdomain systemd[1]: Starting MariaDB database server...
mar 16 21:12:07 localhost.localdomain mariadb-prepare-db-dir[2106]: Database MariaDB is probably...
mar 16 21:12:07 localhost.localdomain mariadb-prepare-db-dir[2106]: If this is not the case, mak...
mar 16 21:12:07 localhost.localdomain mysqld_safe[2140]: 230316 21:12:07 mysqld_safe Logging to...
mar 16 21:12:07 localhost.localdomain mysqld_safe[2140]: 230316 21:12:07 mysqld_safe Starting m...ql
mar 16 21:12:09 localhost.localdomain systemd[1]: Started MariaDB database server.
Hint: Some lines were ellipsized, use -l to show in full.
[root@localhost ~]# _
```

Verifique que el estado del servidor sea **active (running)**

- Ahora terminamos con las tareas de instalación de mariadb ejecutando el comando **mysql_secure_installation**

```
[root@localhost ~]# mysql_secure_installation


NOTE: RUNNING ALL PARTS OF THIS SCRIPT IS RECOMMENDED FOR ALL MariaDB
SERVERS IN PRODUCTION USE! PLEASE READ EACH STEP CAREFULLY!

In order to log into MariaDB to secure it, we'll need the current
password for the root user. If you've just installed MariaDB, and
you haven't set the root password yet, the password will be blank,
so you should just press enter here.
```

Luego de este paso aparecerá una ventana que nos pedirá ingresar el password del root en caso de reinstalación, pero como esta es una instalación nueva no poseemos contraseña así que damos **enter**.

```
Enter current password for root (enter for none):
OK, successfully used password, moving on...
```

Ahora nos preguntará si deseamos colocarle un password al root, aquí definiremos la contraseña del usuario superadministrador del gestor de bases de datos, es esencial seguir una política de contraseña robusta y guardarla en un lugar seguro para usarla

 <p>SISTEMA AUTONOMO DE INFORMACION AUTOMOTRIZ</p>	<p>SAIA</p> <p>Informe de migración de datos</p>	<p>SENA</p>
---	--	--------------------

cuando es debido.


```
Set root password? [Y/n] Y
New password:
Re-enter new password:
Password updated successfully!
Reloading privilege tables..
... Success!
```

Acto siguiente se nos preguntara sobre la remoción de usuarios anónimos, es importante que no se permitan este tipo de accesos a la base de datos así que daremos Yes a este comentario.

```
Remove anonymous users? [Y/n] Y
... Success!
```

```
Normally, root should only be allowed to connect from 'localhost'. This
ensures that someone cannot guess at the root password from the network.
```

Luego nos preguntará si deseamos deshabilitar el conectarnos de forma remota

 <p>SISTEMA AUTÓNOMO DE INFORMACIÓN AUTOMOTRIZ</p>	<p>SAIA</p> <p>Informe de migración de datos</p>	<p>SENA</p>
---	--	-------------

utilizando el root, por seguridad no se debe permitir esto, así que daremos Yes.

```
Normally, root should only be allowed to connect from 'localhost'. This
ensures that someone cannot guess at the root password from the network.

Disallow root login remotely? [Y/n] Y
... Success!
```

Removemos la base de datos test ya que no la necesitaremos

```
Remove test database and access to it? [Y/n] Y
- Dropping test database...
... Success!
- Removing privileges on test database...
... Success!
```

Reiniciamos los privilegios de las tablas y ya estaremos listos para usar el gestor

```
Reloading the privilege tables will ensure that all changes made so
will take effect immediately.

Reload privilege tables now? [Y/n] Y
... Success!

Cleaning up...


All done! If you've completed all of the above steps, your MariaDB
installation should now be secure.

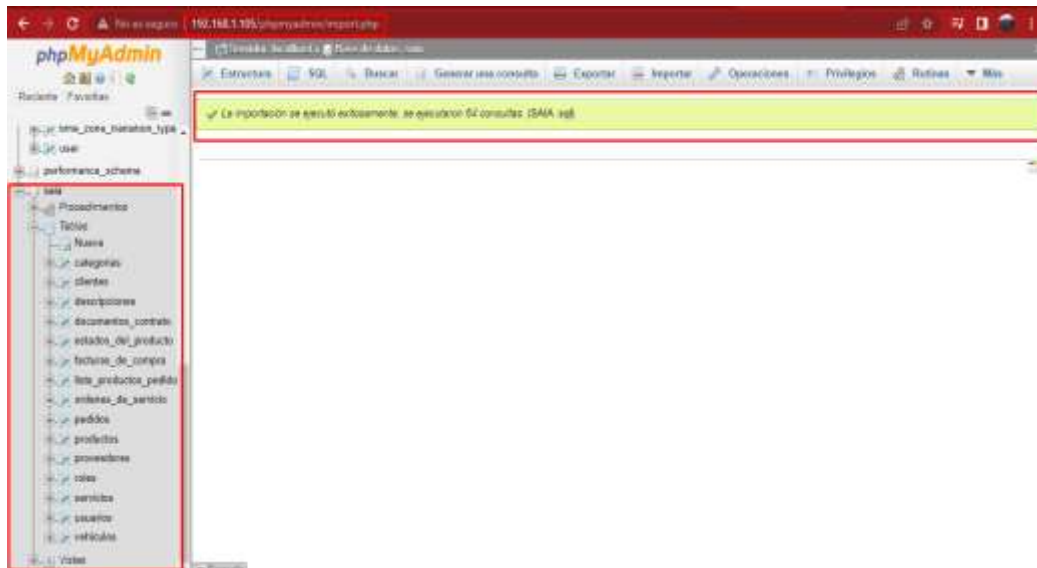
Thanks for using MariaDB!
```

2.3 Instalación de nuestra base de datos en MariaDB.

Con el directorio y archivos completos de nuestra aplicación, entregaremos un fichero .sql llamado saia.sql exportado directamente desde nuestra bd funcional que contendrá la estructura de datos con compatibilidad para mariadb. Para instalarlo en el gestor puede hacerlo de 2 formas, si posee phpmyadmin importe la base de datos directamente al gestor, en caso de no poseer phpmyadmin. Hagalo usando el comando myqlump de mariadb hacia su base de datos

Con phpmyadmin

 <p>SISTEMA AUTÓNOMO DE INFORMACIÓN AUTOMOTRIZ</p>	<p>SAIA</p> <p>Informe de migración de datos</p>	<p>SENA</p>
---	--	--------------------



Con mysqldump


Pruebe el acceso a mariadb

```
[root@localhost ~]# mysql -u root -p
Enter password:
Welcome to the MariaDB monitor.  Commands end with ; or \g.
Your MariaDB connection id is 12
Server version: 5.5.68-MariaDB MariaDB Server

Copyright (c) 2000, 2018, Oracle, MariaDB Corporation Ab and others.
Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.
MariaDB [(none)]>
```

Cree la base de datos saia

```
MariaDB [(none)]> create database saia;
Query OK, 1 row affected (0.00 sec)
```

	<p style="text-align: center;">SAIA</p> <p style="text-align: center;">Informe de migración de datos</p>	<p style="text-align: center;">SENA</p>
---	--	--

En la misma ruta donde tiene el fichero saia.sql en el servidor ejecute el comando **mysqldump -u root -p saia < saia.sql**


```
[root@localhost ~]# mysqldump -u root -p saia < saia.sql
Enter password:
-- MySQL dump 10.14  Distrib 5.5.68-MariaDB, for Linux (x86_64)
--
-- Host: localhost    Database: saia
--
-- Server version      5.5.68-MariaDB

/*!40101 SET @OLD_CHARACTER_SET_CLIENT=@@CHARACTER_SET_CLIENT */;
/*!40101 SET @OLD_CHARACTER_SET_RESULTS=@@CHARACTER_SET_RESULTS */;
/*!40101 SET @OLD_COLLATION_CONNECTION=@@COLLATION_CONNECTION */;
/*!40101 SET NAMES utf8 */;
/*!40103 SET @OLD_TIME_ZONE=@@TIME_ZONE */;
/*!40103 SET TIME_ZONE='+00:00' */;
/*!40014 SET @OLD_UNIQUE_CHECKS=@@UNIQUE_CHECKS, UNIQUE_CHECKS=0 */;
/*!40014 SET @OLD_FOREIGN_KEY_CHECKS=@@FOREIGN_KEY_CHECKS, FOREIGN_KEY_CHECKS=0 */;
/*!40101 SET @OLD_SQL_MODE=@@SQL_MODE, SQL_MODE='NO_AUTO_VALUE_ON_ZERO' */;
/*!40111 SET @OLD_SQL_NOTES=@@SQL_NOTES, SQL_NOTES=0 */;
/*!40103 SET TIME_ZONE=@OLD_TIME_ZONE */;

/*!40101 SET SQL_MODE=@OLD_SQL_MODE */;
/*!40014 SET FOREIGN_KEY_CHECKS=@OLD_FOREIGN_KEY_CHECKS */;
/*!40014 SET UNIQUE_CHECKS=@OLD_UNIQUE_CHECKS */;
/*!40101 SET CHARACTER_SET_CLIENT=@OLD_CHARACTER_SET_CLIENT */;
/*!40101 SET CHARACTER_SET_RESULTS=@OLD_CHARACTER_SET_RESULTS */;
/*!40101 SET COLLATION_CONNECTION=@OLD_COLLATION_CONNECTION */;
/*!40111 SET SQL_NOTES=@OLD_SQL_NOTES */;

-- Dump completed on 2023-03-16 21:57:23
[root@localhost ~]#
```

A este punto ya tendrá lista la base de datos dentro del gestor lista para ser utilizada por el sistema.

 <p>SISTEMA AUTÓNOMO DE INFORMACIÓN AUTOMOTRIZ</p>	<p>SAIA</p> <p>Informe de migración de datos</p>	<p>SENA</p>
---	--	-------------

3 Plan de respaldo

3.1 Dumps periódicos

Realizar copias de respaldo periódicas es vital para asegurar que en caso de daños en la base de datos siempre podremos recuperar la integridad de la información en un porcentaje cercano al 100%, una de las formas más fáciles de respaldar nuestra base de datos es crear backups periódicos de la misma, al estar trabajando en sistemas operativos Windows lo podemos lograr con un script sencillo que nos ayudara a que cuando este sea ejecutado se cree el backup de la siguiente manera.

- Primero necesitaremos crear un fichero para no pasar nuestra contraseña de la bd en texto plano, para este ejemplo lo llamaremos my.cnf y lo guardaremos en el directorio root, en el colocaremos los datos de usuario y contraseña de acceso a la BD.

Se recomienda crear un usuario único para la gestión de la base de datos y con privilegios solo a esta.

```
GNU nano 2.3.1 Fichero: my.cnf
[mysqldump]
user=saia
password=xxxxxxx
```

Cambie los datos según corresponda.

- Cree un directorio donde se guardaran los respaldos, para el ejemplo creamos el directorio backupBD en la carpeta /root/


```
[root@localhost ~]# mkdir backupBD
```

- Utilice el comando nano (o del editor de su preferencia) para crear el script bash necesario.

Nano backupBD.sh

Inicialmente creamos unas variables de tiempo para poder anexarlas a los archivos que se crearan periódicamente, esto con el fin de poder distinguirlos de uno a otro dependiendo el caso.

Seguido a ello configure los demás ítems como nombre de la bd, ruta del archivo my.cnf que guarda los accesos a la bd, ruta de salida de los backups, el nombre del archivo de respaldo con las variables de tiempo y el comando a ejecutar.

 <p>SISTEMA AUTÓNOMO DE INFORMACIÓN AUTOMOTRIZ</p>	<p>SAIA</p> <p>Informe de migración de datos</p>	<p>SENA</p>
---	--	-------------

```
#!/bin/bash

dia=`date +%d-%m-%Y`
hora=`date +%H:%M`

# Respaldo base de datos de MySQL con
# mysqldump y archivo de configuración
NOMBRE_BASE_DE_DATOS="saia"
RUTA_ARCHIVO_CONFIGURACION="/root/my.cnf"
RUTA_SALIDA_RESPALDO="/root/backupBD"
NOMBRE_ARCHIVO_RESPALDO="saia_${dia}.${hora}.sql"
RUTA_ARCHIVO_RESPALDO="${RUTA_SALIDA_RESPALDO}/${NOMBRE_ARCHIVO_RESPALDO}"
mysqldump --defaults-file=${RUTA_ARCHIVO_CONFIGURACION} ${NOMBRE_BASE_DE_DATOS} > ${RUTA_ARCHIVO_RESPALDO}
```

Este fichero nos creará un respaldo de la base de datos en la ruta /root/backupBD/ con la sintaxis de nombre de archivo **saia_”dia de realización del backup”.”Hora de realización”.sql**

Guardamos y le damos los permisos necesarios para poder ser ejecutado como script y estará listo para ejecutar.

```
[root@localhost ~]# chmod +x backupBD.sh
```

Ejecutamos

```
[root@localhost ~]# ./backupBD.sh
```


Verificamos que el archivo se haya creado

```
[root@localhost backupBD]# pwd
/root/backupBD
[root@localhost backupBD]# ls
saia 17-03-2023.21:25.sql
[root@localhost backupBD]#
```

Y comprobaremos que el backup se creó en la ruta y a la fecha y hora según corresponda, para terminar, crearemos una tarea programada en cron y estará lista nuestra base de datos para ser respaldada automáticamente.

Abrimos crontab utilizando el editor de texto de preferencia y escribimos las líneas necesarias para ejecutar el script a diario o según lo disponga el administrador.

nano /etc/crontab

	<p style="text-align: center;">SAIA</p> <p style="text-align: center;">Informe de migración de datos</p>	<p style="text-align: center;">SENA</p>
---	--	--

```

GNU nano 2.3.1                               Fichero: /etc/crontab


SHELL=/bin/bash
PATH=/sbin:/bin:/usr/sbin:/usr/bin
MAILTO=root

# For details see man 4 crontabs

# Example of job definition:
# .----- minute (0 - 59)
# | .----- hour (0 - 23)
# | | .----- day of month (1 - 31)
# | | | .----- month (1 - 12) OR jan,feb,mar,apr ...
# | | | | .---- day of week (0 - 6) (Sunday=0 or 7) OR sun,mon,tue,wed,thu,fri,sat
# | | | | |
# * * * * * user-name command to be executed
30 23 * * * root /root/backupBD.sh

```

Aquí le decimos a crontrab que ejecute el script backupBD.sh todos los días a las 11:30 PM cuando el tráfico es bajo.

 <p>SISTEMA AUTÓNOMO DE INFORMACIÓN AUTOMOTRIZ</p>	<p>SAIA</p> <p>Informe de migración de datos</p>	<p>SENA</p>
---	--	-------------

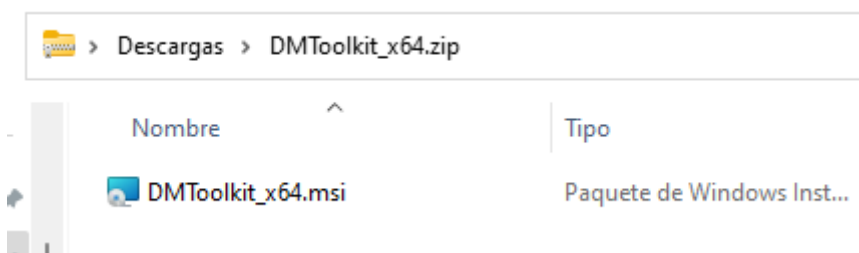
4 Plan de migración

4.1 Migración a SQL server.

Para migrar nuestra base de datos utilizaremos un programa sencillo gratuito hasta los 50.000 registros que inicialmente para nuestra base de datos será más que suficiente, este software se llama ESF – Database Migration toolkit – pro para descargarlo ingrese al siguiente enlace.

https://www.dbsofts.com/es/articles/mariadb_a_sql_server/

descargar un zip que luego de descomprimirlo será un archivo msi y basta con ejecutarlo e instalarlo en la maquina.




Luego de instalada la herramienta la tendrá lista para utilizar en su escritorio.



Ejecútela e inicie con la migración

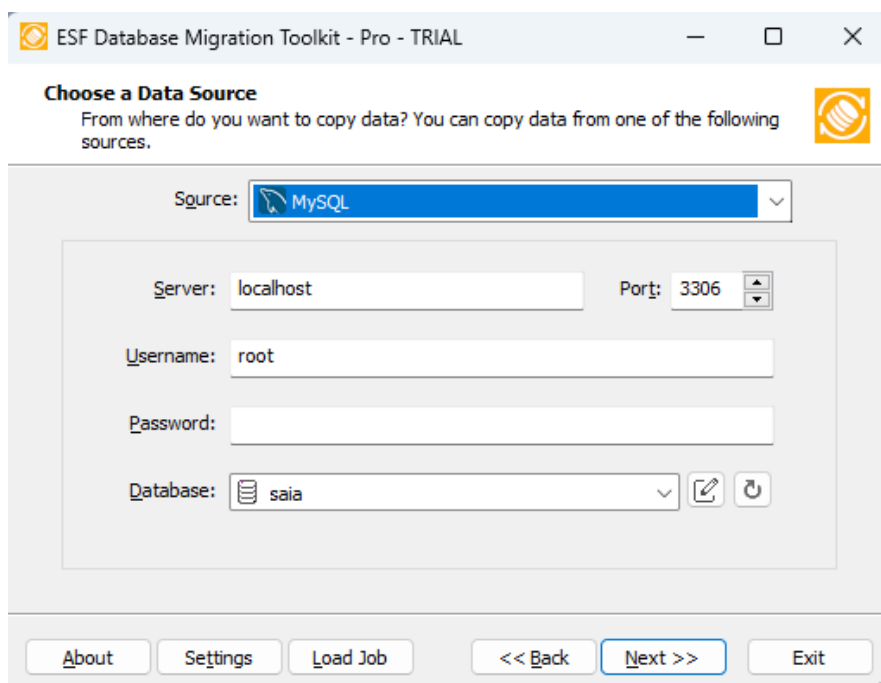
Se iniciará una ventana de asistente que nos dará indicaciones iniciales del programa como notas de que es una versión trial y que a partir de los 50.000 registros debe acceder a la versión full y nos dará un enlace para acceder a ellos


	<p style="text-align: center;">SAIA</p> <p style="text-align: center;">Informe de migración de datos</p>	<p style="text-align: center;">SENA</p>
---	--	--



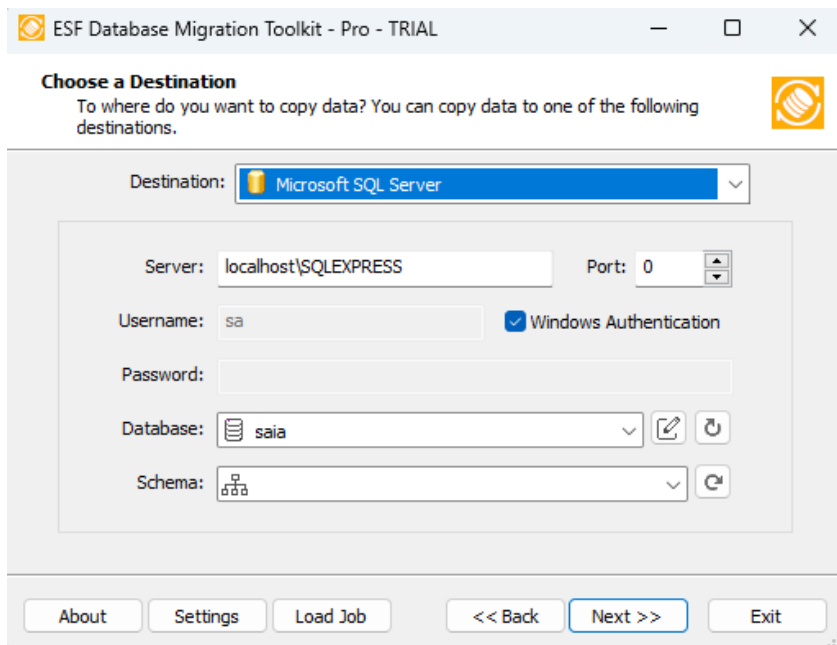
Damos clic en siguiente y nos preguntará el origen de donde queremos migrar la base de datos en esta migración lo haremos de MariaDB – MySQL a SQLserver.

Así que el origen será MySQL, ingrese los datos del gestor y la base de datos a migrar y de siguiente



	<p style="text-align: center;">SAIA</p> <p style="text-align: center;">Informe de migración de datos</p>	<p style="text-align: center;">SENA</p>
---	--	--

Ahora nos exigirá el gestor y la base de datos de destino, seleccionamos SQLserver y lleamos los datos de acceso al SQLserver, en este caso por ser una instalación local lo haremos con la autenticación de Windows, damos clic en siguiente.



Choose a Destination
To where do you want to copy data? You can copy data to one of the following destinations.

Destination: **Microsoft SQL Server**

Server: **localhost\SQLEXPRESS** Port: **0**

Username: **sa** ☒ Windows Authentication

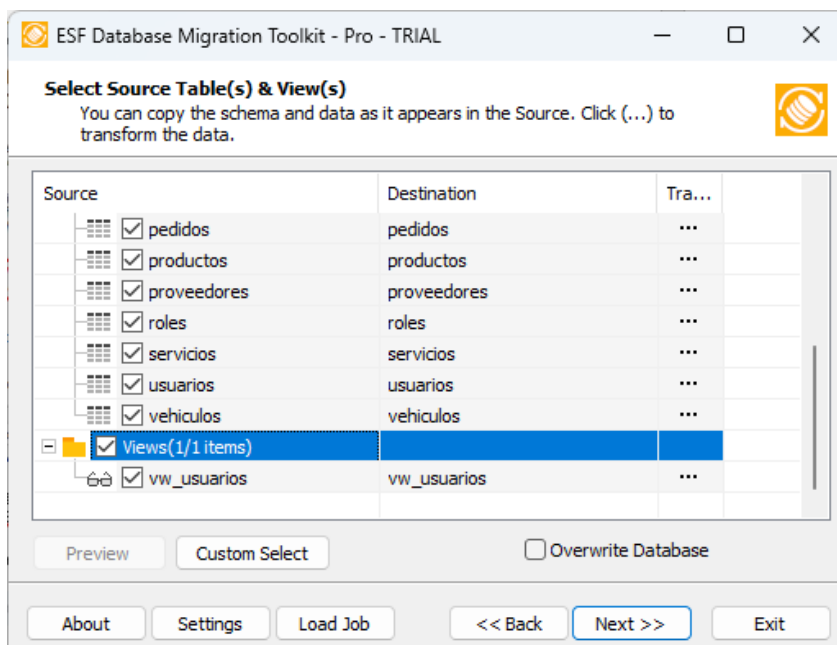
Password:

Database: **saia**

Schema: **data**

Buttons: About, Settings, Load Job, << Back, Next >>, Exit


Ahora seleccionaremos las tablas y las vistas a migrar y daremos siguiente



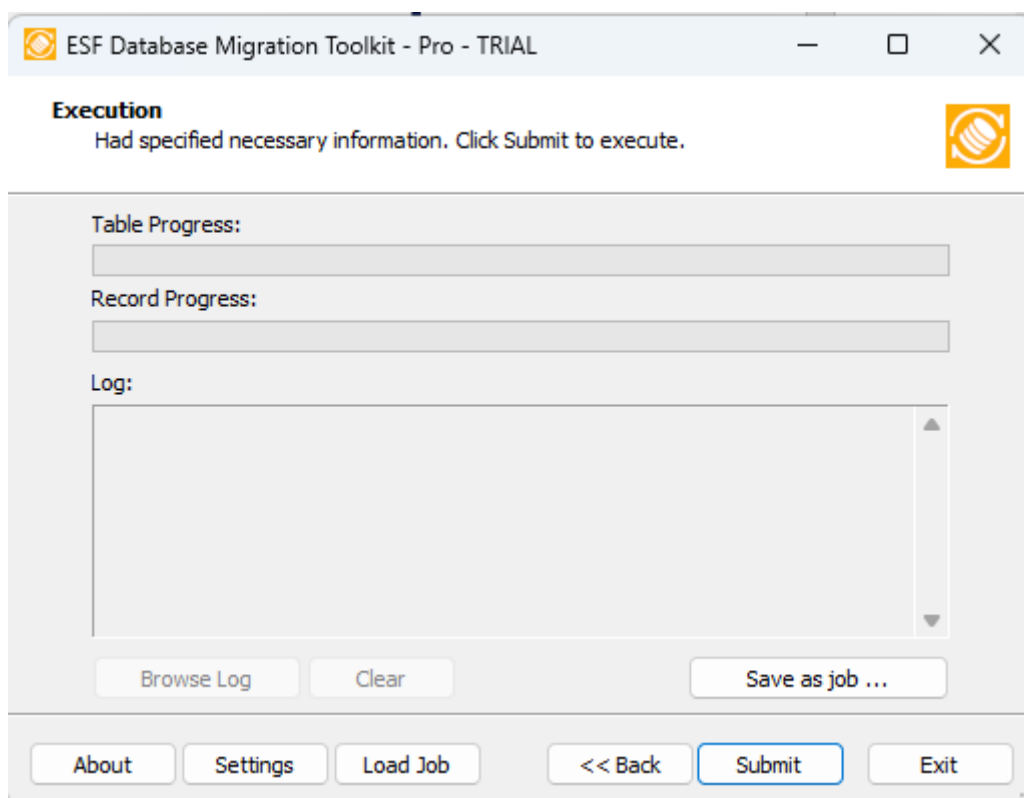
Select Source Table(s) & View(s)
You can copy the schema and data as it appears in the Source. Click (...) to transform the data.

Source	Destination	Tra...
<input checked="" type="checkbox"/> pedidos	pedidos	...
<input checked="" type="checkbox"/> productos	productos	...
<input checked="" type="checkbox"/> proveedores	proveedores	...
<input checked="" type="checkbox"/> roles	roles	...
<input checked="" type="checkbox"/> servicios	servicios	...
<input checked="" type="checkbox"/> usuarios	usuarios	...
<input checked="" type="checkbox"/> vehiculos	vehiculos	...
<input checked="" type="checkbox"/> Views(1/1 items)		
<input checked="" type="checkbox"/> vw_usuarios	vw_usuarios	...

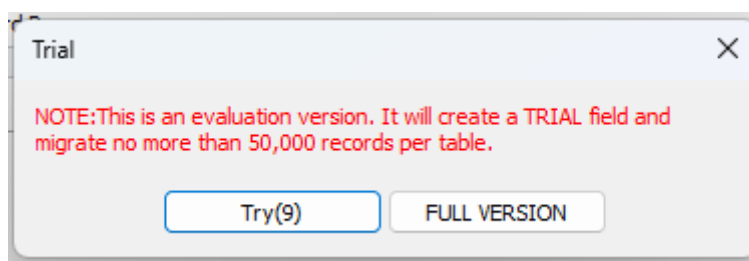
Buttons: Preview, Custom Select, Overwrite Database, About, Settings, Load Job, << Back, Next >>, Exit


	<p style="text-align: center;">SAIA</p> <p style="text-align: center;">Informe de migración de datos</p>	<p style="text-align: center;">SENA</p>
---	--	--

Al dar siguiente nos encontraremos con una pantalla de progreso donde iniciaremos la migración, allí damos Submit .

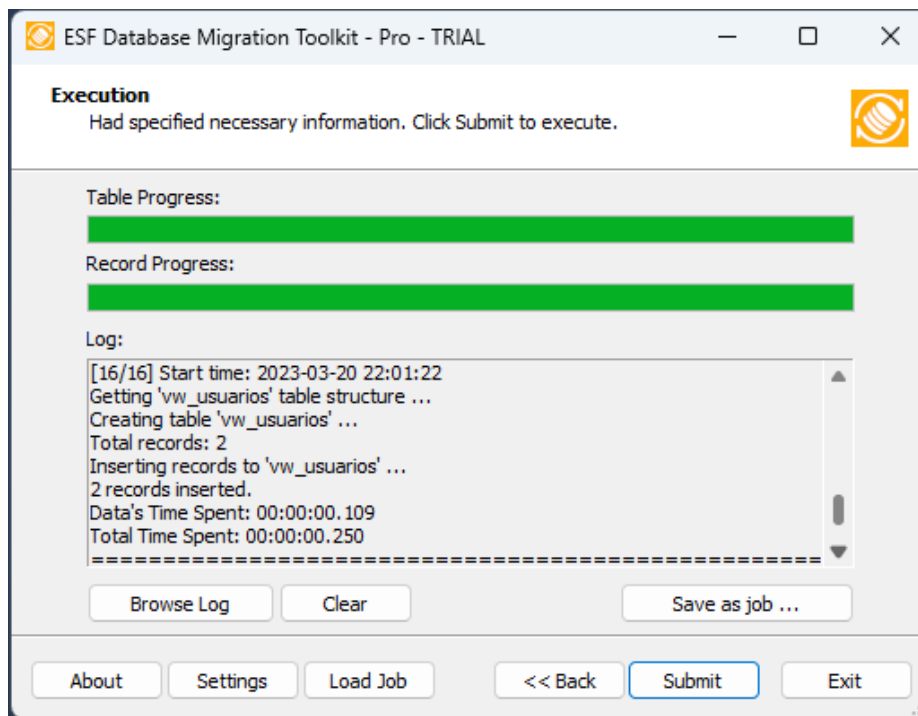


Ahora esperaremos a que pasen los 10 segundos de la versión trial y damos clic para que inicie el proceso

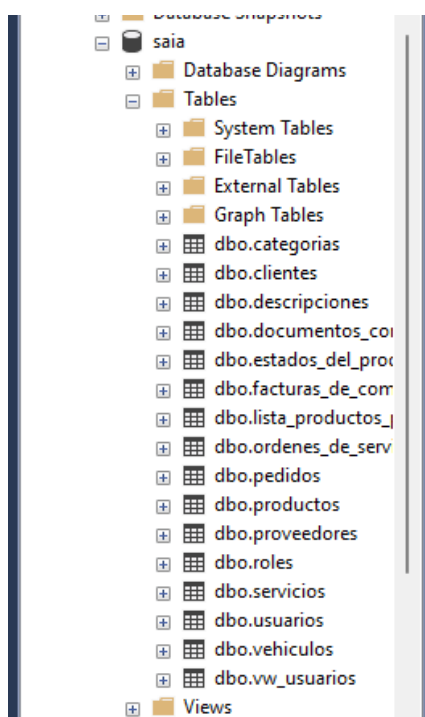


 <p>SISTEMA AUTÓNOMO DE INFORMACIÓN AUTOMOTRIZ</p>	<p>SAIA</p> <p>Informe de migración de datos</p>	<p>SENA</p>
---	--	--------------------

EL proceso iniciara y esperaremos a que la barra de progreso se llene en su totalidad y damos exit para salir del programa.



Ingresamos a SQLserver y encontraremos la base de datos migrada



	<p align="center">SAIA</p> <p align="center">Informe de migración de datos</p>	<p align="center">SENA</p>
--	--	-----------------------------------

Si realizamos un select a las tablas que en mysql tienen contenido, podremos evidenciar que los datos poseen contenido.

Select desde el origen.

✓ Mostrando filas 0 - 1 (total de 2, La consulta tardó 0,0019 segundos.)

```
select * from usuarios;
```

☐ Perfilando [Editar en línea] [Editar] [Explicar SQL] [Crear código PHP] [Actualizar]

☐ Mostrar todo | Número de filas: 25 | Filtrar filas: Buscar en esta tabla | Sort by key: Ninguna

+ Opciones

	documento	apellidos	nombres	email	pass	telefono	foto	id_rol
<input type="checkbox"/> Editar <input type="checkbox"/> Copiar <input type="checkbox"/> Borrar	52534780	prueba	prueba	prueba@prueba.com		3133134564	[BLOB - 31 B]	2
<input type="checkbox"/> Editar <input type="checkbox"/> Copiar <input type="checkbox"/> Borrar	52534789	limas	sergio	sflimas2@misena.edu.co	12345	3133133214	[BLOB - 23 B]	1

☐ Seleccionar todo | Para los elementos que están marcados: ☐ Editar ☐ Copiar ☐ Borrar ☐ Exportar

Select desde el destino.

SQLQuery3.sql - DE...10B5QC\Sergio (65)*


```
select * from usuarios;
```

100 %

Results Messages

	documento	apellidos	nombres	email	pass	telefono	foto	id_rol	TRIAL998
1	52534780	prueba	prueba	prueba@prueba.com		3133134564	0x442E20436C617365732050726F796563746F2053414941...	2	T
2	52534789	limas	sergio	sflimas2@misena.edu.co	12345	3133133214	0x666C6578626F782D6164617074617469766F322E706E67	1	T

A este punto ya tendrá las herramientas necesarias para poder instalar la base de datos en su gestor original MariaDB, Realizar Dumps periódicos como técnica de respaldo y migrar la base de datos a otro gestor como SQLserver.

	<p style="text-align: center;">SAIA</p> <p style="text-align: center;">Informe de migración de datos</p>	<p style="text-align: center;">SENA</p>
---	---	--

5 5. Cambios en el código para SQL Server.

5.1 Conexión a la base de datos con SQL Server.

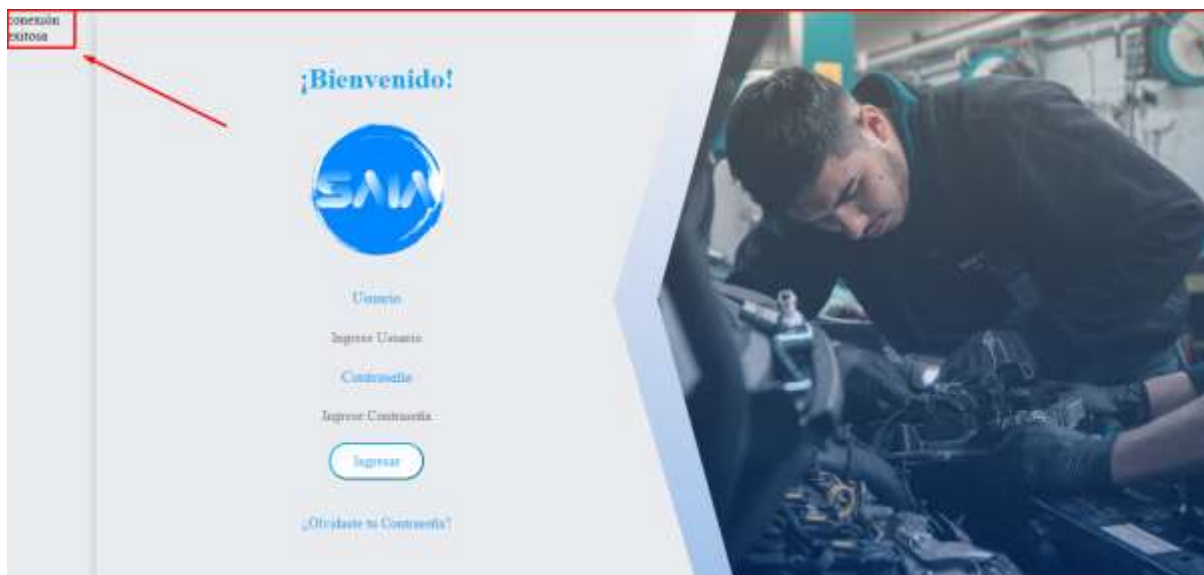
Se cambian los datos de acceso en el archivo de conexión por los datos propios para sql server, esto luego de haber instalado los complementos necesarios de php para SQL server.

```

1  <?php
2      $servername = "DESKTOP-2I0B5QC\SQLEXPRESS";
3      $connectionInfo = array("Database"=>"saia", "UID"=>"", "PWD"=>"", "CharacterSet"=>"UTF-8");
4      $conn = sqlsrv_connect($servername, $connectionInfo);
5
6      if($conn){
7          echo "conexión exitosa";
8      }else {
9          echo "conexión fallida";
10         die (print_r(sqlsrv_errors(), true));
11     }
12

```


Es necesario realizar la prueba de conexión para estar seguros de que funciona correctamente.



Nos saldrá el aviso de conexión exitosa, en caso de que no, nos mostrará algunos logs de error que podemos identificar para solucionar, cuando este seguro de que funciona correctamente comente el echo que habilita la frase conexión exitosa.

5.2 Ingresar roles

Luego de habilitar la conexión creamos los ficheros necesarios para el CRUD de roles, iniciando por el CREATE que llamaremos Ingresar roles, a continuación les mostraremos la sintaxis utilizada para crear el rol en SQL server.

 <p>SISTEMA AUTÓNOMO DE INFORMACIÓN AUTOMOTRIZ</p>	<p>SAIA</p> <p>Informe de migración de datos</p>	<p>SENA</p>
---	--	-------------

```

insert.php
1  <?php
2
3  include("DataBase.php");
4
5
6  if(isset($_POST['insertar'])){
7      $codigo_rol = $_POST['rol_codigo'];
8      $nombre_rol = $_POST['rol_nombre'];
9
10     $insert= "INSERT INTO roles (Nombre) VALUES ('$nombre_rol')";
11
12     $ejecutar= sqlsrv_query($conn, $insert);
13
14     if($ejecutar){
15         echo "<h3>Query enviado con exito</h3>";
16         header("Location:index.php");
17     } else {
18         echo "<h3>No enviado</h3>";
19         die (print_r(sqlsrv_errors(), true));
20     }
21 }
22
23 ?>

```

Probaremos creando el rol Mecanico20

Inicio / Módulo Usuarios / Crear Rol

Crear Rol

Consultar Roles

Código Rol

Código

Nombre Rol


Mecanico20

Enviar

Cerrar

Aparece el rol mecanico20 en la consulta a la bd y deberá aparecernos cuando consultemos desde la aplicación

id	Nombre
6	Mecanico16
7	Mecanico17
8	Mecanico17
9	Mecanico17
10	Mecanico17
11	Mecanico19
12	Mecanico20
13	Mecanico21
14	Mecanico21
15	Mecanico20

	<p style="text-align: center;">SAIA</p> <p style="text-align: center;">Informe de migración de datos</p>	<p style="text-align: center;">SENA</p>
---	---	--

5.3 Consultar Roles

Para consultar los roles utilizamos la siguiente sintaxis.

Dentro de la vista rol_read.view.php en la tabla creada para mostrar los registros de la BD, creamos la siguiente sintaxis.

```

<?php
include("DataBase.php");

$consulta = "SELECT * FROM roles";

$ejecutar = sqlsrv_query($conn, $consulta);

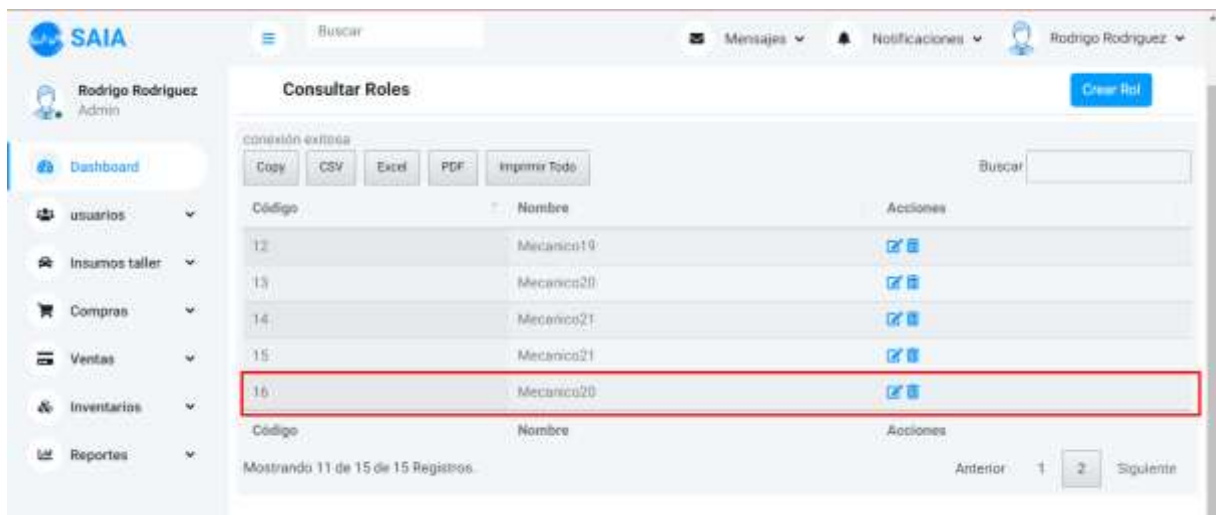
$i=0;

while($fila = sqlsrv_fetch_array($ejecutar)){
    $codigo_rol = $fila['id'];
    $nombre_rol = $fila['Nombre'];
    $i++;

    <tr>
        <td><?php echo $codigo_rol ?></td>
        <td><?php echo $nombre_rol ?></td>
        <td class="tabla-acciones">
            <a class="tabla-edit" href="rol_read.view.php?editar=<?php echo $codigo_rol ?>"><i class="fas fa-edit"></i></a>
            <a class="tabla-delete" href="rol_read.view.php?eliminar=<?php echo $codigo_rol ?>"><i class="fas fa-trash-alt"></i></a>
        </td>
    </tr>
}
<?php
}
?>

```

En la vista nos aparecerán los registros de la tabla roles entre ellos el mecanico20 que agregamos anteriormente.



Código	Nombre	Acciones
12	Mecanico10	Edit Delete
13	Mecanico20	Edit Delete
14	Mecanico21	Edit Delete
15	Mecanico21	Edit Delete
16	Mecanico20	Edit Delete