



**Taller de Base de Datos: Fidelización y Análisis de Colaboradores**

**Euris J. Rodríguez V. 8-1013-2315**

**Objetivo**

Este taller tiene como objetivo principal diseñar e implementar una base de datos relacional para la empresa XYZ, enfocándose en la gestión de usuarios, perfiles, y un sistema de fidelización de colaboradores. Se hará un énfasis especial en el uso de vistas SQL para facilitar el análisis de datos y la toma de decisiones gerenciales.

**Criterios y Requisitos**

El sistema de base de datos deberá soportar los siguientes módulos y funcionalidades:

1. Módulos Principales: Usuarios, Perfiles, Fidelización, Login. Se deben considerar tablas intermedias según sea necesario.
2. Gestión de Colaboradores (Usuarios):
  - Cada usuario debe tener los atributos: nombre, apellido, estado (activo/inactivo), contraseña, cargo, salario, fecha\_ingreso, y un perfil asociado.
  - Un usuario debe poder tener un solo perfil activo a la vez.
3. Sistema de Login:
  - La base de datos debe registrar cada inicio de sesión de cada usuario en una tabla de Login, incluyendo la fecha\_hora\_login y el estado\_login (exitoso/fallido).



4. Módulo "Fideliza a tu Personal":

- Este módulo gestionará los registros de usuarios basados en su participación en actividades de la empresa.
- Cada 15 días, la empresa realiza actividades donde los colaboradores participan y acumulan puntos. Se debe registrar la fecha\_actividad, el tipo\_actividad, la descripción\_actividad y los puntos\_otorgados por actividad.
- La información de fidelización debe estar directamente relacionada con los perfiles y usuarios. Un usuario obtiene puntos según su participación en actividades, y estos puntos pueden influir en su perfil o estatus de fidelización.

5. Perfiles de Usuarios:

- Los perfiles están basados en roles y deben contener: nombre\_perfil, fecha\_vigencia\_perfil, descripción\_perfil, y un encargado\_perfil (puede ser un usuario que gestiona ese perfil o un rol específico).
- Se debe implementar la capacidad de que un perfil tenga varios permisos asociados, aunque no se pide crear la tabla de permisos en este taller, se debe prever su futura integración.



I PARTE: Simulación de Datos Mínimo (40 Pts):

- 20 Usuarios
- 10 Perfiles diferentes
- 100 Registros de autenticación (Login)
- Registros de actividades de fidelización para 12 meses (simular al menos 2 actividades por mes).
- Calcular la cantidad de puntos de fidelización acumulados por cada usuario en 12 meses.

-Crear contenedor y que este corriendo en segundo plan:

```
PS C:\Users\jw818> docker run -d --name mariadb_taller2 -e MARIADB_ROOT_PASS=WORD=123 -p 3307:3306 mariadb
5e109ee3c4749438cd184244c558ed18649e0be064901eb8250bf063b6afcccd8
PS C:\Users\jw818> docker exec -it mariadb_taller2 bash
root@5e109ee3c474:/# mariadb -u root -p
Enter password:
```

```
PS C:\Users\jw818> docker ps
CONTAINER ID IMAGE COMMAND CREATED STATUS
PORTS NAMES
5e109ee3c474 mariadb "docker-entrypoint.s..." 19 minutes ago Up 19 minutes
0.0.0.0:3307->3306/tcp, [::]:3307->3306/tcp mariadb_taller2
PS C:\Users\jw818>
```



Base de datos creada:

```
MariaDB [(none)]> SHOW DATABASES;
+-----+
| Database |
+-----+
| information_schema |
| mysql |
| performance_schema |
| sys |
| taller2 |
+-----+
5 rows in set (0.001 sec)

MariaDB [(none)]> CREATE DATABASE taller_2;
Query OK, 1 row affected (0.003 sec)

MariaDB [(none)]> SHOW DATABASES;
+-----+
| Database |
+-----+
| information_schema |
| mysql |
| performance_schema |
| sys |
| taller2 |
| taller_2 |
+-----+
6 rows in set (0.001 sec)

MariaDB [(none)]>
```



Seleccionando la base de datos:

```
MariaDB [(none)]> use taller_2;
Database changed
MariaDB [taller_2]>
```

Tabla de perfiles:

```
MariaDB [taller_2]> CREATE TABLE perfiles (
    ->     id_perfil INT AUTO_INCREMENT PRIMARY KEY,
    ->     nombre_perfil VARCHAR(50) NOT NULL,
    ->     descripcion_perfil VARCHAR(200),
    ->     fecha_vigencia_perfil DATE,
    ->     encargado_perfil VARCHAR(100)
    -> );
Query OK, 0 rows affected (0.026 sec)

MariaDB [taller_2]> |
```

```
MariaDB [taller_2]> DESCRIBE perfiles;
+-----+-----+-----+-----+-----+
| Field          | Type       | Null | Key | Default | Extra
+-----+-----+-----+-----+-----+
| id_perfil      | int(11)    | NO   | PRI | NULL    | auto_increme
nt |
| nombre_perfil  | varchar(50) | NO   |     | NULL    | |
| descripcion_perfil | varchar(200) | YES  |     | NULL    | |
| fecha_vigencia_perfil | date      | YES  |     | NULL    | |
| encargado_perfil | varchar(100) | YES  |     | NULL    | |
+-----+-----+-----+-----+-----+
---+
5 rows in set (0.002 sec)

MariaDB [taller_2]>
```



Creando la tabla usuarios:

```
MariaDB [taller_2]> CREATE TABLE usuarios (
->     id_usuario INT AUTO_INCREMENT PRIMARY KEY,
->     nombre VARCHAR(50) NOT NULL,
->     apellido VARCHAR(50) NOT NULL,
->     estado ENUM('activo','inactivo') NOT NULL,
->     contrasena VARCHAR(255) NOT NULL,
->     cargo VARCHAR(100),
->     salario DECIMAL(10,2),
->     fecha_ingreso DATE,
->     id_perfil INT NOT NULL,
->     CONSTRAINT fk_usuarios_perfiles
->         FOREIGN KEY (id_perfil) REFERENCES perfiles(id_perfil)
-> );
Query OK, 0 rows affected (0.066 sec)

MariaDB [taller_2]> DESCRIBE usuarios;
+-----+-----+-----+-----+-----+
| Field          | Type           | Null | Key | Default | Extra
+-----+-----+-----+-----+-----+
| id_usuario      | int(11)        | NO   | PRI | NULL    | auto_in
crement |
| nombre          | varchar(50)    | NO   |     | NULL    | 
| apellido        | varchar(50)    | NO   |     | NULL    | 
| estado          | enum('activo','inactivo') | NO   |     | NULL    | 
| contrasena      | varchar(255)   | NO   |     | NULL    | 
| cargo           | varchar(100)   | YES  |     | NULL    | 
| salario          | decimal(10,2)  | YES  |     | NULL    | 
| fecha_ingreso   | date           | YES  |     | NULL    | 
| id_perfil       | int(11)        | NO   | MUL | NULL    | 
+-----+-----+-----+-----+-----+
9 rows in set (0.002 sec)

MariaDB [taller_2]> |
```



Creando login:

```
4 rows in set (0.002 sec)

MariaDB [taller_2]> CREATE TABLE login (
->     id_login INT AUTO_INCREMENT PRIMARY KEY,
->     id_usuario INT NOT NULL,
->     fecha_hora_login DATETIME NOT NULL,
->     estado_login ENUM('exitoso','fallido') NOT NULL,
->     CONSTRAINT fk_login_usuarios
->         FOREIGN KEY (id_usuario) REFERENCES usuarios(id_usuario)
-> );
Query OK, 0 rows affected (0.053 sec)

MariaDB [taller_2]> DESCRIBE login;
+-----+-----+-----+-----+-----+
| Field          | Type           | Null | Key | Default | Extra
+-----+-----+-----+-----+-----+
| id_login       | int(11)        | NO   | PRI | NULL    | auto
| increment      |                |
| id_usuario     | int(11)        | NO   | MUL | NULL    | 
|                  |
| fecha_hora_login | datetime      | NO   |      | NULL    | 
|                  |
| estado_login   | enum('exitoso','fallido') | NO   |      | NULL    | 
+-----+-----+-----+-----+-----+
4 rows in set (0.002 sec)

MariaDB [taller_2]>
```



Creando tabla actividades\_fidelizacion:

```
MariaDB [taller_2]> CREATE TABLE actividades_fidelizacion (
->     id_actividad INT AUTO_INCREMENT PRIMARY KEY,
->     fecha_actividad DATE NOT NULL,
->     tipo_actividad VARCHAR(50),
->     descripcion_actividad VARCHAR(200),
->     puntos_otorgados INT NOT NULL
-> );
Query OK, 0 rows affected (0.033 sec)

MariaDB [taller_2]> DESCRIBE actividades_fidelizacion;
+-----+-----+-----+-----+-----+
| Field          | Type       | Null | Key | Default | Extra
+-----+-----+-----+-----+-----+
| id_actividad   | int(11)    | NO  | PRI | NULL    | auto_increme
nt |
| fecha_actividad | date      | NO  |     | NULL    | |
| tipo_actividad  | varchar(50) | YES |     | NULL    | |
| descripcion_actividad | varchar(200) | YES |     | NULL    | |
| puntos_otorgados | int(11)    | NO  |     | NULL    | |
+-----+-----+-----+-----+-----+
---+
5 rows in set (0.002 sec)

MariaDB [taller_2]> |
```



Creando la tabla de usuarios\_actividades:

```
MariaDB [taller_2]> CREATE TABLE usuarios_actividades (
->     id_usuario INT NOT NULL,
->     id_actividad INT NOT NULL,
->     PRIMARY KEY (id_usuario, id_actividad),
->     CONSTRAINT fk_ua_usuario
->         FOREIGN KEY (id_usuario) REFERENCES usuarios(id_usuario),
->     CONSTRAINT fk_ua_actividad
->         FOREIGN KEY (id_actividad) REFERENCES actividades_fidelizacion(id_actividad)
-> );
Query OK, 0 rows affected (0.024 sec)

MariaDB [taller_2]> DESCRIBE usuarios_actividades;
+-----+-----+-----+-----+-----+
| Field      | Type   | Null | Key  | Default | Extra |
+-----+-----+-----+-----+-----+
| id_usuario | int(11) | NO   | PRI   | NULL    |       |
| id_actividad | int(11) | NO   | PRI   | NULL    |       |
+-----+-----+-----+-----+-----+
2 rows in set (0.002 sec)

MariaDB [taller_2]>
```

Chequeando tablas:

```
MariaDB [taller_2]> SHOW TABLES;
+-----+
| Tables_in_taller_2 |
+-----+
| actividades_fidelizacion |
| login |
| perfiles |
| usuarios |
| usuarios_actividades |
+-----+
5 rows in set (0.001 sec)

MariaDB [taller_2]>
```



Insertando perfiles:

```
MariaDB [taller_2]> INSERT INTO perfiles (nombre_perfil, descripcion_perfil,  
fecha_vigencia_perfil, encargado_perfil) VALUES  
    -> ('Administrador', 'Acceso total al sistema', '2024-01-01', 'Gerencia'),  
    -> ('Supervisor', 'Supervisión de equipos', '2024-01-01', 'Gerencia'),  
    -> ('Analista', 'Análisis de información', '2024-01-01', 'Supervisor'),  
    -> ('Recursos Humanos', 'Gestión del personal', '2024-01-01', 'Gerencia'),  
    -> ('Tecnología', 'Soporte y desarrollo TI', '2024-01-01', 'Administrador')  
  
    -> ('Ventas', 'Gestión comercial', '2024-01-01', 'Supervisor'),  
    -> ('Marketing', 'Publicidad y campañas', '2024-01-01', 'Gerencia'),  
    -> ('Finanzas', 'Control financiero', '2024-01-01', 'Gerencia'),  
    -> ('Logística', 'Distribución y operaciones', '2024-01-01', 'Supervisor'),  
  
    -> ('Operativo', 'Operaciones generales', '2024-01-01', 'Supervisor');  
Query OK, 10 rows affected (0.008 sec)  
Records: 10  Duplicates: 0  Warnings: 0
```



```
MariaDB [taller_2]> SELECT * FROM perfiles;
+-----+-----+-----+
| id_perfil | nombre_perfil | descripcion_perfil | fecha_vigencia_perfil | encargado_perfil |
+-----+-----+-----+
| 1 | Administrador | Acceso total al sistema | 2024-01-01 | Gerencia |
| 2 | Supervisor | Supervisión de equipos | 2024-01-01 | Gerencia |
| 3 | Analista | Análisis de información | 2024-01-01 | Supervisor |
| 4 | Recursos Humanos | Gestión del personal | 2024-01-01 | Gerencia |
| 5 | Tecnología | Soporte y desarrollo TI | 2024-01-01 | Administrador |
| 6 | Ventas | Gestión comercial | 2024-01-01 | Supervisor |
| 7 | Marketing | Publicidad y campañas | 2024-01-01 | Gerencia |
| 8 | Finanzas | Control financiero | 2024-01-01 | Gerencia |
| 9 | Logística | Distribución y operaciones | 2024-01-01 | Supervisor |
| 10 | Operativo | Operaciones generales | 2024-01-01 | Supervisor |
+-----+-----+-----+
-----+
10 rows in set (0.001 sec)

MariaDB [taller_2]>
```



Insertando usuarios:

```
MariaDB [taller_2]> INSERT INTO usuarios
-> (nombre, apellido, estado, contrasena, cargo, salario, fecha_ingreso
id_perfil) VALUES
-> ('Juan', 'Perez', 'activo', '123', 'Analista', 1200, '2023-01-10', 3),
-> ('Ana', 'Gomez', 'activo', '123', 'RRHH', 1500, '2022-03-15', 4),
-> ('Luis', 'Martinez', 'activo', '123', 'Soporte TI', 1800, '2021-06-20', 5),
-> ('Maria', 'Lopez', 'activo', '123', 'Vendedora', 1100, '2023-02-01', 6),
-> ('Carlos', 'Diaz', 'activo', '123', 'Marketing', 1300, '2022-08-10', 7),
-> ('Sofia', 'Ruiz', 'activo', '123', 'Finanzas', 1700, '2021-11-05', 8),
-> ('Pedro', 'Castro', 'activo', '123', 'Logística', 1250, '2022-05-19', 9),
-> ('Laura', 'Mendoza', 'activo', '123', 'Operativo', 1000, '2023-04-01', 10),
-> ('Diego', 'Ortega', 'activo', '123', 'Supervisor', 1600, '2020-09-15', 2),
-> ('Valeria', 'Ramos', 'activo', '123', 'Analista', 1200, '2023-06-12', 3),
-> ('Miguel', 'Torres', 'activo', '123', 'Ventas', 1150, '2022-02-20', 6),
-> ('Andrea', 'Morales', 'activo', '123', 'Marketing', 1350, '2021-10-10', 7),
-> ('Fernando', 'Navarro', 'activo', '123', 'IT', 1750, '2020-03-03', 5),
-> ('Paola', 'Vega', 'activo', '123', 'RRHH', 1500, '2022-12-01', 4),
-> ('Ricardo', 'Pineda', 'activo', '123', 'Finanzas', 1650, '2021-07-07', 8),
-> ('Natalia', 'Santos', 'activo', '123', 'Logística', 1300, '2023-01-25', 9),
-> ('Oscar', 'Herrera', 'activo', '123', 'Operativo', 1050, '2022-06-18', 10),
-> ('Daniela', 'Flores', 'activo', '123', 'Ventas', 1120, '2023-03-30', 6),
-> ('Jorge', 'Molina', 'activo', '123', 'Supervisor', 1550, '2020-01-20', 2),
-> ('Camila', 'Reyes', 'activo', '123', 'Analista', 1250, '2023-09-01', 3);
Query OK, 20 rows affected (0.033 sec)
Records: 20  Duplicates: 0  Warnings: 0
```



Revisando los datos:

```
MariaDB [taller_2]> SELECT id_usuario, nombre, apellido, id_perfil FROM usuarios;
+-----+-----+-----+-----+
| id_usuario | nombre | apellido | id_perfil |
+-----+-----+-----+-----+
| 1 | Juan | Perez | 3 |
| 2 | Ana | Gomez | 4 |
| 3 | Luis | Martinez | 5 |
| 4 | Maria | Lopez | 6 |
| 5 | Carlos | Diaz | 7 |
| 6 | Sofia | Ruiz | 8 |
| 7 | Pedro | Castro | 9 |
| 8 | Laura | Mendoza | 10 |
| 9 | Diego | Ortega | 2 |
| 10 | Valeria | Ramos | 3 |
| 11 | Miguel | Torres | 6 |
| 12 | Andrea | Morales | 7 |
| 13 | Fernando | Navarro | 5 |
| 14 | Paola | Vega | 4 |
| 15 | Ricardo | Pineda | 8 |
| 16 | Natalia | Santos | 9 |
| 17 | Oscar | Herrera | 10 |
| 18 | Daniela | Flores | 6 |
| 19 | Jorge | Molina | 2 |
| 20 | Camila | Reyes | 3 |
+-----+-----+-----+-----+
20 rows in set (0.001 sec)

MariaDB [taller_2]>
```



Insertnado en actividades\_fidelizacion:

```
MariaDB [taller_2]> INSERT INTO actividades_fidelizacion
-> (fecha_actividad, tipo_actividad, descripcion_actividad, puntos_otorgados) VALUES
-> ('2024-01-15','Capacitación','Curso interno','50'),
-> ('2024-01-30','Team Building','Actividad grupal','40'),
-> ('2024-02-15','Capacitación','Curso técnico','50'),
-> ('2024-02-28','Reconocimiento','Empleado del mes','60'),
-> ('2024-03-15','Capacitación','Curso liderazgo','50'),
-> ('2024-03-30','Integración','Actividad recreativa','40'),
-> ('2024-04-15','Capacitación','Curso ventas','50'),
-> ('2024-04-30','Reconocimiento','Buen desempeño','60'),
-> ('2024-05-15','Capacitación','Curso finanzas','50'),
-> ('2024-05-30','Integración','Dinámica grupal','40'),
-> ('2024-06-15','Capacitación','Curso TI','50'),
-> ('2024-06-30','Reconocimiento','Empleado destacado','60'),
-> ('2024-07-15','Capacitación','Curso marketing','50'),
-> ('2024-07-30','Integración','Actividad recreativa','40'),
-> ('2024-08-15','Capacitación','Curso logística','50'),
-> ('2024-08-30','Reconocimiento','Buen desempeño','60'),
-> ('2024-09-15','Capacitación','Curso RRHH','50'),
-> ('2024-09-30','Integración','Actividad grupal','40'),
-> ('2024-10-15','Capacitación','Curso operaciones','50'),
-> ('2024-10-30','Reconocimiento','Empleado del mes','60'),
-> ('2024-11-15','Capacitación','Curso liderazgo','50'),
-> ('2024-11-30','Integración','Team building','40'),
-> ('2024-12-15','Capacitación','Curso anual','50'),
-> ('2024-12-30','Reconocimiento','Cierre de año','60');
Query OK, 24 rows affected (0.036 sec)
Records: 24  Duplicates: 0  Warnings: 0
```



```
MariaDB [taller_2]> SELECT COUNT(*) FROM actividades_fidelizacion;
+-----+
| COUNT(*) |
+-----+
|      24 |
+-----+
1 row in set (0.035 sec)

MariaDB [taller_2]> INSERT INTO usuarios_actividades (id_usuario, id_actividad)
-> SELECT u.id_usuario, a.id_actividad
-> FROM usuarios u
-> JOIN actividades_fidelizacion a
-> ON a.id_actividad % u.id_usuario = 0;
Query OK, 80 rows affected (0.039 sec)
Records: 80  Duplicates: 0  Warnings: 0

MariaDB [taller_2]> SELECT COUNT(*) FROM usuarios_actividades;
+-----+
| COUNT(*) |
+-----+
|      80 |
+-----+
1 row in set (0.001 sec)

MariaDB [taller_2]>
```

Insert en login

```
MariaDB [taller_2]> INSERT INTO login (id_usuario, fecha_hora_login, estado_login)
-> SELECT
->     FLOOR(1 + RAND()*20),
->     DATE_ADD('2024-01-01', INTERVAL FLOOR(RAND()*365) DAY),
->     IF(RAND() > 0.25,'exitoso','fallido')
-> FROM dual
-> LIMIT 100;
Query OK, 1 row affected (0.035 sec)
Records: 1  Duplicates: 0  Warnings: 0

MariaDB [taller_2]> SELECT COUNT(*) FROM login;
+-----+
| COUNT(*) |
+-----+
|      1 |
+-----+
1 row in set (0.001 sec)

MariaDB [taller_2]>
```



**Universidad  
Interamericana de Panamá**

**LAUREATE INTERNATIONAL UNIVERSITIES®**

**II PARTE: Aplicación y uso de las Vistas (60 Pts):**

Se requiere implementar las siguientes vistas:

1. Vista de Desempeño de Colaboradores:

- Cree una vista llamada v\_DesempenoColaboradores que muestre para cada usuario:
  - nombre\_completo
  - cargo
  - salario
  - fecha\_ingreso
  - total\_puntos\_fidelizacion\_acumulados (sumatoria de todos los puntos ganados)
  - promedio\_puntos\_por\_actividad
  - estado\_fidelizacion (un campo calculado basado en la siguiente lógica: 'Excelente' si el total\_puntos\_fidelizacion\_acumulados es mayor a 500, 'Bueno' si está entre 200 y 500, 'Regular' si es menor a 200).
  - dias\_desde\_ultimo\_login (días transcurridos desde su último inicio de sesión exitoso).



CREANDO VISTA:

```
MariaDB [taller_2]> CREATE OR REPLACE VIEW v_DesempenoColaboradores AS
-> SELECT
->     CONCAT(u.nombre, ' ', u.apellido) AS nombre_completo,
->     u.cargo,
->     u.salario,
->     u.fecha_ingreso,
->     COALESCE(SUM(af.puntos_otorgados), 0) AS total_puntos_fidelizacion
->     _acumulados,
->     AVG(af.puntos_otorgados) AS promedio_puntos_por_actividad,
->     CASE
->         WHEN COALESCE(SUM(af.puntos_otorgados), 0) > 500 THEN 'Excelente'
->         WHEN COALESCE(SUM(af.puntos_otorgados), 0) BETWEEN 200 AND 500
->         THEN 'Bueno'
->         ELSE 'Regular'
->     END AS estado_fidelizacion,
->     DATEDIFF(
->         CURRENT_DATE,
->         MAX(l.fecha_hora_login)
->     ) AS dias_desde_ultimo_login
-> FROM usuarios u
-> LEFT JOIN usuarios_actividades ua ON u.id_usuario = ua.id_usuario
-> LEFT JOIN actividades_fidelizacion af ON ua.id_actividad = af.id_actividad
-> LEFT JOIN login l
->     ON u.id_usuario = l.id_usuario
->     AND l.estado_login = 'exitoso'
-> GROUP BY u.id_usuario;
Query OK, 0 rows affected (0.049 sec)
```

## VIENDO LA VISTA:

```
MariaDB [taller_2]> SELECT * FROM v_DesempenoColaboradores;
+-----+-----+-----+-----+-----+-----+
| nombre_completo | cargo | salario | fecha_ingreso | total_puntos_fidelizacion_acumulados | pro medio_puntos_por_actividad | estado_fidelizacion | dias_desde_ultimo_login |
+-----+-----+-----+-----+-----+-----+
| Juan Perez | Analista | 1200.00 | 2023-01-10 | NULL | 1200 |
| Ana Gomez | RRHH | 50.0000 | Excelente | 50.0000 | 600 |
| Luis Martinez | Soporte TI | 1800.00 | 2021-06-20 | 50.0000 | 400 |
| Maria Lopez | Vendedora | 1100.00 | 2023-02-01 | 60.0000 | 360 |
| Carlos Diaz | Marketing | 1300.00 | 2022-08-10 | 50.0000 | 200 |
| Sofia Ruiz | Finanzas | 1700.00 | 2021-11-05 | 50.0000 | 200 |
| Pedro Castro | Logística | 1250.00 | 2022-05-19 | 46.6667 | 140 |
| Laura Mendoza | Operativo | 1000.00 | 2023-04-01 | 60.0000 | 180 |
| Diego Ortega | Supervisor | 1600.00 | 2020-09-15 | 45.0000 | 90 |
| Valeria Ramos | Analista | 1200.00 | 2023-06-12 | 50.0000 | 100 |
| Miguel Torres | Ventas | 1150.00 | 2022-02-20 | 45.0000 | 90 |
| Andrea Morales | Marketing | 1350.00 | 2021-10-10 | 60.0000 | 120 |
| Fernando Navarro | IT | 1750.00 | 2020-03-03 | 50.0000 | 50 |
| Paola Vega | RRHH | 1500.00 | 2022-12-01 | 40.0000 | 40 |
| Ricardo Pineda | Finanzas | 1650.00 | 2021-07-07 | 50.0000 | 50 |
| Natalia Santos | Logística | 1300.00 | 2023-01-25 | 60.0000 | 60 |
| Oscar Herrera | Operativo | 1050.00 | 2022-06-18 | 50.0000 | 50 |
| Daniela Flores | Ventas | 1120.00 | 2023-03-30 | 40.0000 | 40 |
| Jorge Molina | Supervisor | 1550.00 | 2020-01-20 | 50.0000 | 50 |
| Camila Reyes | Analista | 1250.00 | 2023-09-01 | 60.0000 | 60 |
+-----+-----+-----+-----+-----+-----+
20 rows in set (0.006 sec)
```



2. Vista de Actividades por Perfil:

- Cree una vista llamada v\_actividadesPorPerfil que muestre la distribución de la participación en actividades por cada tipo de perfil. Para cada perfil, debe incluir:
  - nombre\_perfil
  - descripcion\_perfil
  - cantidad\_usuarios\_con\_este\_perfil
  - total\_actividades\_participadas\_por\_perfil (sumatoria de participaciones de usuarios con ese perfil en actividades)
  - promedio\_puntos\_por\_usuario\_en\_este\_perfil.
  - porcentaje\_participacion\_total (porcentaje de actividades en las que participaron usuarios con este perfil, respecto al total de actividades realizadas por la empresa).

VISTA 2:

```
MariaDB [taller_2]> CREATE OR REPLACE VIEW v_actividadesPorPerfil AS
-> SELECT
->     p.nombre_perfil,
->     p.descripcion_perfil,
->     COUNT(DISTINCT u.id_usuario) AS cantidad_usuarios_con_este_perfil,
->     COUNT(ua.id_actividad) AS total_actividades_participadas_por_perfil,
->     AVG(af.puntos_otorgados) AS promedio_puntos_por_usuario_en_este_perfil,
->     ROUND(
->         (COUNT(ua.id_actividad) /
->          (SELECT COUNT(*) FROM actividades_fidelizacion)) * 100, 2
->     ) AS porcentaje_participacion_total
-> FROM perfiles p
-> LEFT JOIN usuarios u ON p.id_perfil = u.id_perfil
-> LEFT JOIN usuarios_actividades ua ON u.id_usuario = ua.id_usuario
-> LEFT JOIN actividades_fidelizacion af ON ua.id_actividad = af.id_actividad
-> GROUP BY p.id_perfil;
Query OK, 0 rows affected (0.026 sec)
```

```
MariaDB [taller_2]> SELECT * FROM v_actividadesPorPerfil;
+-----+-----+-----+-----+
| nombre_perfil | descripcion_perfil | cantidad_usuarios_con_este_perfil | total_actividades_participadas_por_perfil | promedio_puntos_por_usuario_en_este_perfil | porcentaje_participacion_total |
+-----+-----+-----+-----+
| Administrador | Acceso total al sistema | NULL | 0 | 0.00 |
| Supervisor | Supervisión de equipos | 46.6667 | 2 | 12.50 |
| Analista | Análisis de información | 50.3704 | 3 | 112.50 |
| Recursos Humanos | Gestión del personal | 49.2308 | 2 | 54.17 |
| Tecnología | Soporte y desarrollo TI | 50.0000 | 2 | 37.50 |
| Ventas | Gestión comercial | 54.4444 | 3 | 37.50 |
| Marketing | Publicidad y campañas | 53.3333 | 2 | 25.00 |
| Finanzas | Control financiero | 50.0000 | 2 | 20.83 |
| Logística | Distribución y operaciones | 50.0000 | 2 | 16.67 |
| Operativo | Operaciones generales | 57.5000 | 2 | 16.67 |
+-----+-----+-----+-----+
10 rows in set (0.002 sec)

MariaDB [taller_2]>
```

### 3. Vista de Historial de Login Detallado:

- Cree una vista llamada v\_historialLoginDetallado que muestre por cada intento de login:
  - nombre\_usuario
  - apellido\_usuario
  - cargo\_usuario
  - fecha\_hora\_login
  - estado\_login
  - tiempo\_desde\_anterior\_login (diferencia en minutos o segundos desde el login inmediatamente anterior del mismo usuario, si existe). Si es el primer login, este campo puede ser NULL.



CREANDO VISTA:

```
MariaDB [taller_2]> CREATE OR REPLACE VIEW v_historialLoginDetallado AS
-> SELECT
->     u.nombre AS nombre_usuario,
->     u.apellido AS apellido_usuario,
->     u.cargo AS cargo_usuario,
->     l.fecha_hora_login,
->     l.estado_login,
->     TIMESTAMPDIFF(
->         MINUTE,
->         LAG(l.fecha_hora_login)
->             OVER (PARTITION BY u.id_usuario ORDER BY l.fecha_hora_login),
->         l.fecha_hora_login
->     ) AS tiempo_desde_anterior_login
-> FROM login l
-> JOIN usuarios u ON l.id_usuario = u.id_usuario;
Query OK, 0 rows affected (0.041 sec)
```

```
MariaDB [taller_2]> SELECT * FROM v_historialLoginDetallado;
+-----+-----+-----+-----+-----+
| nombre_usuario | apellido_usuario | cargo_usuario | fecha_hora_login | estado_login | tiempo_desde_anterior_login |
+-----+-----+-----+-----+-----+
| Paola        | Vega           | RRHH          | 2024-11-29 00:00:00 | exitoso      |
+-----+-----+-----+-----+-----+
1 row in set (0.020 sec)

MariaDB [taller_2]>
```

VERIFICANDO VISTAS:

```
MariaDB [taller_2]> SHOW FULL TABLES WHERE TABLE_TYPE = 'VIEW';
+-----+-----+
| Tables_in_taller_2 | Table_type |
+-----+-----+
| v_DesempenoColaboradores | VIEW    |
| v_actividadesPorPerfil | VIEW    |
| v_historialLoginDetallado | VIEW    |
+-----+-----+
3 rows in set (0.002 sec)

MariaDB [taller_2]>
```

4. Uso de Vistas para Escenarios de Negocio:

- Demuestre cómo utilizar las vistas creadas para responder a las siguientes preguntas gerenciales, proporcionando las consultas SQL:
  - ¿Cuáles son los 5 colaboradores con mejor desempeño en fidelización en el último trimestre y cuál es su cargo?

```
MariaDB [taller_2]> SELECT
    ->     nombre_completo,
    ->     cargo,
    ->     total_puntos_fidelizacion_acumulados
    -> FROM v_DesempenoColaboradores
    -> ORDER BY total_puntos_fidelizacion_acumulados DESC
    -> LIMIT 5;
+-----+-----+-----+
| nombre_completo | cargo      | total_puntos_fidelizacion_acumulados |
+-----+-----+-----+
| Juan Perez     | Analista   | 1200          |
| Ana Gomez      | RRHH       | 600           |
| Luis Martinez  | Soporte TI | 400           |
| Maria Lopez    | Vendedora  | 360           |
| Carlos Diaz    | Marketing  | 200           |
+-----+-----+-----+
5 rows in set (0.003 sec)

MariaDB [taller_2]>
```

Esta consulta permite identificar a los colaboradores con mayor acumulación de puntos de fidelización, facilitando decisiones de reconocimiento e incentivos.

- ¿Qué perfiles tienen la menor participación en actividades de fidelización y requieren un plan de incentivos?

```
MariaDB [taller_2]> SELECT
->     nombre_perfil,
->     total_actividades_participadas_por_perfil,
->     porcentaje_participacion_total
->   FROM v_actividadesPorPerfil
-> ORDER BY total_actividades_participadas_por_perfil ASC;
+-----+-----+-----+
| nombre_perfil | total_actividades_participadas_por_perfil | porcentaje_participacion_total |
+-----+-----+-----+
| Administrador | 0 | 0.00 |
| Supervisor | 3 | 12.50 |
| Logística | 4 | 16.67 |
| Operativo | 4 | 16.67 |
| Finanzas | 5 | 20.83 |
| Marketing | 6 | 25.00 |
| Tecnología | 9 | 37.50 |
| Ventas | 9 | 37.50 |
| Recursos Humanos | 13 | 54.17 |
| Analista | 27 | 112.50 |
+-----+-----+-----+
10 rows in set (0.002 sec)

MariaDB [taller_2]>
```

Los perfiles con menor participación representan oportunidades de mejora mediante estrategias de motivación y fidelización.

- ¿Qué usuarios no han iniciado sesión en los últimos 30 días y cuál fue su último cargo?

```
MariaDB [taller_2]> SELECT
->     nombre_completo,
->     cargo,
->     dias_desde_ultimo_login
->   FROM v_DesempenoColaboradores
-> WHERE dias_desde_ultimo_login > 30;
+-----+-----+-----+
| nombre_completo | cargo | dias_desde_ultimo_login |
+-----+-----+-----+
| Paola Vega | RRHH | 436 |
+-----+-----+-----+
1 row in set (0.003 sec)
```

Esta consulta permite identificar posibles usuarios inactivos o con bajo nivel de compromiso con el sistema.

- Obtener un reporte mensual de la cantidad de logins exitosos vs. fallidos.

Documentación del Proceso

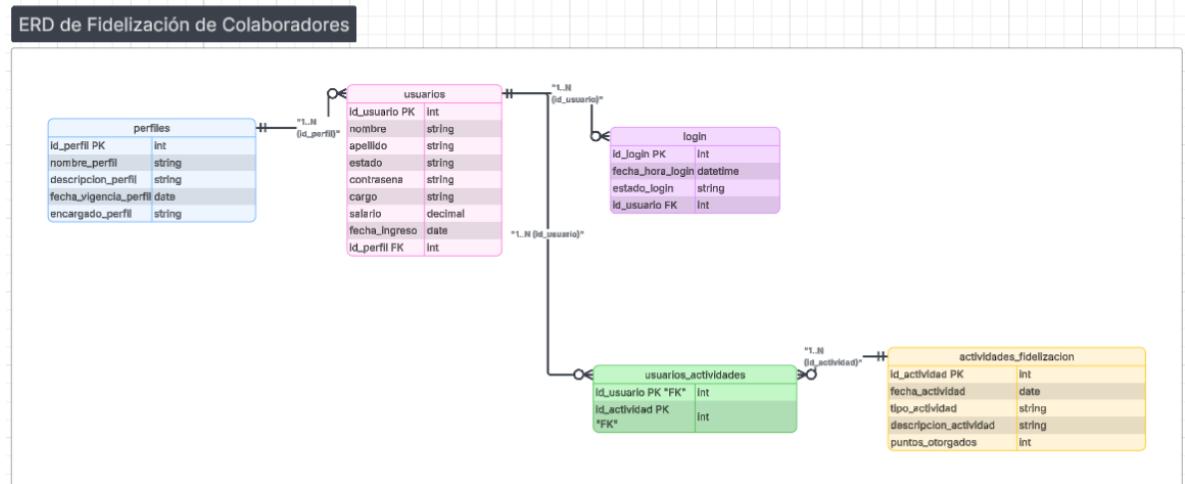
```
MariaDB [taller_2]> SELECT
->      MONTH(fecha_hora_login) AS mes,
->      estado_login,
->      COUNT(*) AS cantidad
->  FROM login
-> GROUP BY mes, estado_login
-> ORDER BY mes;
+-----+-----+-----+
| mes | estado_login | cantidad |
+-----+-----+-----+
|   11 | exitoso       |       1 |
+-----+-----+-----+
1 row in set (0.002 sec)

MariaDB [taller_2]>
```

Este reporte permite evaluar el comportamiento de acceso al sistema y detectar posibles problemas de autenticación.

El entregable debe incluir un documento detallado que abarque:

- Diagrama Entidad-Relación (ERD)



- Scripts SQL Completos:

- Creación de la base de datos y tablas.
- Sentencias INSERT para poblar la base de datos con los datos de simulación
- requeridos.
- Definición de todas las vistas solicitadas.
- Consultas SQL que demuestren el uso de las vistas para responder a los
- escenarios de negocio.

- Lecciones Aprendidas: En este apartado deben documentar los desafíos encontrados, las soluciones implementadas y el aprendizaje obtenido, especialmente en el diseño y uso de vistas.



## SCRIPTS SQL

### **Creación de la Base de Datos y Tablas**

Se desarrollaron scripts SQL para la creación de la base de datos y sus tablas principales, considerando los módulos solicitados: usuarios, perfiles, login y fidelización.

Las tablas fueron definidas con sus respectivas claves primarias y foráneas, garantizando la integridad referencial entre los datos y permitiendo una correcta relación entre usuarios, perfiles y actividades de fidelización.

El diseño de las tablas respeta los principios de normalización, evitando redundancia de información y facilitando la escalabilidad futura del sistema.

### **Sentencias INSERT para Datos de Simulación**

Se implementaron sentencias INSERT para poblar la base de datos con datos simulados, cumpliendo los requisitos mínimos establecidos en el taller:

- Creación de 20 usuarios con información completa.
- Definición de 10 perfiles distintos.
- Registro de al menos 100 intentos de inicio de sesión.
- Registro de actividades de fidelización durante un período de 12 meses, simulando un mínimo de dos actividades por mes.
- Asignación de puntos de fidelización a los usuarios según su participación en actividades.



### **Definición de las Vistas SQL**

Se desarrollaron todas las vistas solicitadas en el enunciado con el objetivo de facilitar el análisis de información y la toma de decisiones gerenciales.

Las vistas encapsulan consultas complejas que integran información de múltiples tablas, permitiendo acceder a resultados consolidados sin afectar directamente las tablas base.

Entre las vistas creadas se incluyen aquellas orientadas al desempeño de colaboradores, participación en actividades por perfil y el historial detallado de inicio de sesión.

### **Consultas SQL para Escenarios de Negocio**

Finalmente, se implementaron consultas SQL que utilizan las vistas creadas para responder a distintos escenarios de negocio, tales como:

- Identificación de los colaboradores con mejor desempeño en fidelización.
- Detección de perfiles con baja participación en actividades.
- Identificación de usuarios inactivos en el sistema.
- Análisis de intentos de inicio de sesión exitosos y fallidos por período.

Estas consultas demuestran el uso práctico de las vistas como herramientas de apoyo para el análisis gerencial y la toma de decisiones estratégicas.



## **LECCIONES APRENDIDAS**

Durante el desarrollo de este taller se obtuvieron aprendizajes relevantes tanto a nivel técnico como conceptual, especialmente en el diseño de bases de datos relacionales y el uso de vistas SQL como herramienta de análisis.

### **Importancia de un buen diseño del modelo de datos**

Uno de los principales aprendizajes fue que un diseño adecuado del Diagrama Entidad-Relación facilita significativamente la implementación de la base de datos. Definir correctamente las entidades, atributos, claves primarias y foráneas permitió evitar redundancia de datos y garantizar la integridad referencial entre las tablas.

### **Desafíos en la normalización y relaciones**

Durante el diseño surgieron retos al modelar relaciones de muchos a muchos, lo cual requirió la creación de tablas intermedias. Este proceso permitió comprender la importancia de la normalización para mantener una base de datos escalable, ordenada y fácil de mantener.

### **Uso estratégico de vistas SQL**

El uso de vistas representó un aprendizaje clave, ya que permitieron encapsular consultas complejas y reutilizarlas para distintos escenarios de negocio. Gracias a las vistas, se simplificó el análisis de información sin necesidad de acceder directamente a las tablas base, mejorando la claridad y seguridad de las consultas.

### **Optimización de consultas para escenarios de negocio**

La construcción de consultas basadas en vistas permitió responder de forma eficiente a preguntas gerenciales como desempeño de colaboradores, participación en actividades y niveles de inactividad. Esto evidenció cómo las vistas facilitan la toma de decisiones al presentar información ya procesada y relevante.



**Universidad  
Interamericana de Panamá**

**LAUREATE INTERNATIONAL UNIVERSITIES®**

### **Integración entre datos operativos y analíticos**

Se aprendió a integrar datos de uso operativo, como registros de inicio de sesión, con información analítica relacionada con desempeño y fidelización. Esta integración permitió obtener una visión más completa del comportamiento de los usuarios dentro del sistema.

### **Buenas prácticas en documentación técnica**

Finalmente, se reforzó la importancia de documentar correctamente cada fase del proyecto, incluyendo scripts SQL, diagramas y explicaciones. Una documentación clara no solo facilita la evaluación académica, sino que también mejora la comprensión del sistema por parte de otros desarrolladores o analistas.

- Subir al repositorio en carpeta de TALLER\_03