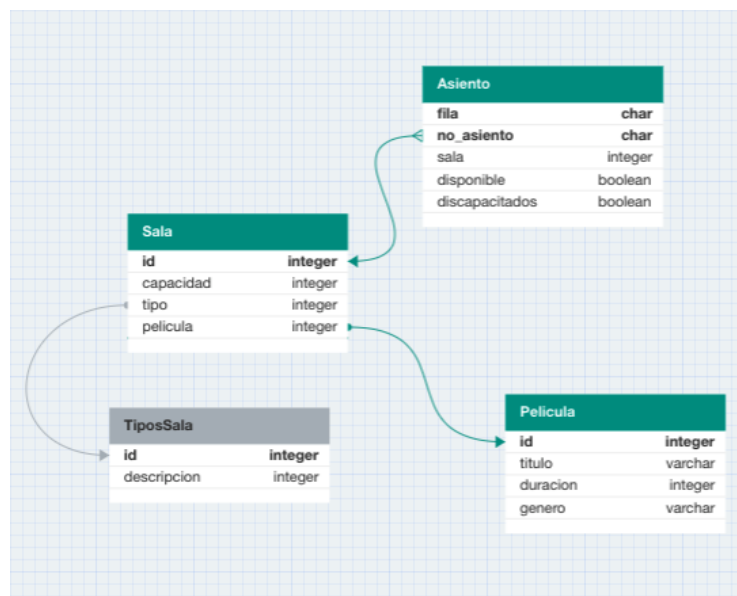
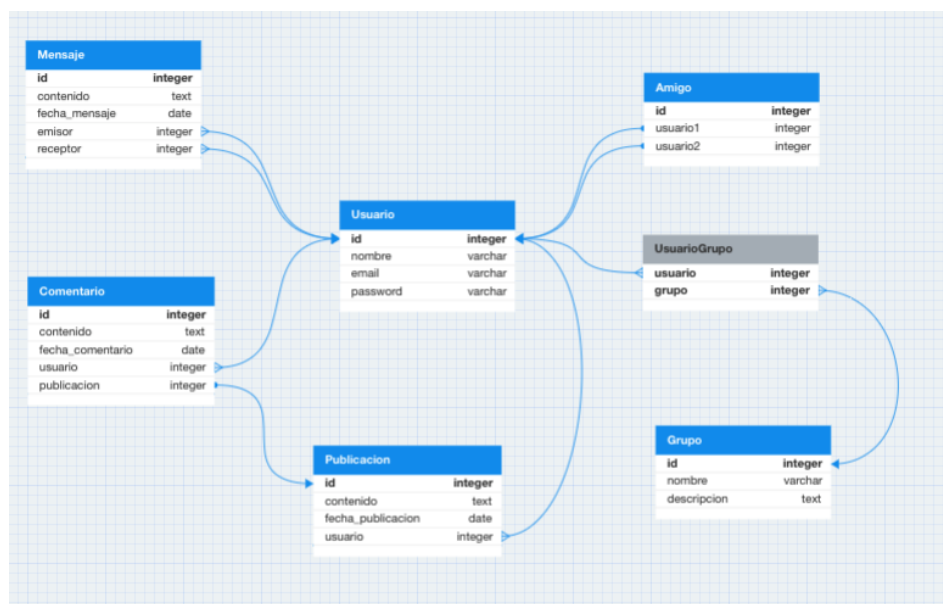


**Carreón González Saúl Horacio  
Martínez Buenrostro Jorge Rafael  
Morales Hernández José Manuel**

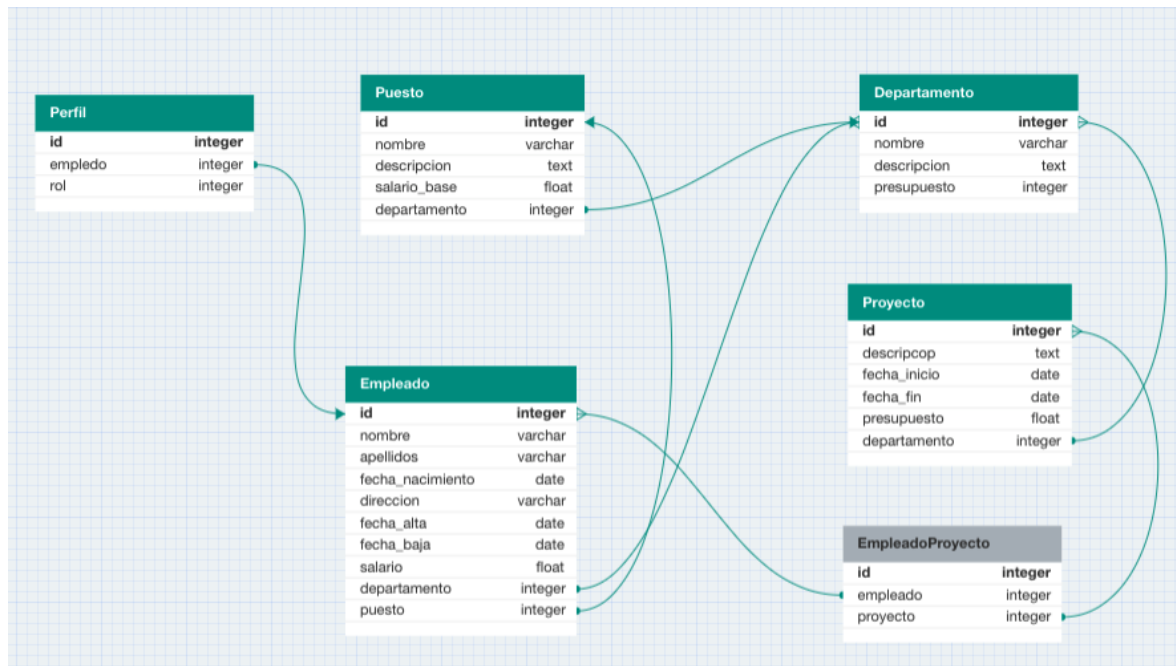
1. Identifique 3 entidades que podrían ser importantes para un cine. Liste al menos 3 atributos para cada una de esas entidades. Muestre el correspondiente diagrama de entidad relación



2. Diseñe un diagrama Entidad-Relación para una red social. Utilice como base las entidades: Usuario, Publicación, Comentario, Amigo. Agregue al menos otras 2 entidades.



3. Diseñe un diagrama Entidad-Relación de acuerdo al sistema elegido en clase usando al menos 5 entidades.



4. Cree el script bd\_practica.sql en el cual se cree la base de datos de acuerdo con el diseño propuesto en el inciso anterior. Además, en la creación de cada tabla se debe tomar en cuenta la relación mostrada en el diagrama (uno a uno, uno a muchos, o muchos a muchos).

```
[MariaDB [EDBM]> show tables;
+-----+
| Tables_in_EDBM |
+-----+
| Departamento   |
| Empleado       |
| EmpleadoProyecto |
| Perfil         |
| Proyecto       |
| Puesto         |
+-----+
6 rows in set (0.000 sec)
```

5. Agregue al bd\_practica.sql las instrucciones necesarias para que se llene la base de datos. Entre más datos mejor para hacer pruebas después.

5. Agregue las restricciones de borrado “ON DELETE CASCADE” Y “ON DELETE RESTRICT” en las tablas que considere necesario.

### RESTRICCIONES ON DELETE CASCADE

- **Relación entre EmpleadoProyecto y Empleado:** Si se elimina un empleado se eliminan todas las asignaciones a proyectos asociadas a ese empleado. Por lo tanto, al eliminar un empleado, también se eliminarán automáticamente las asignaciones de proyectos asociadas a ese empleado.

- **Relación entre EmpleadoProyecto y Proyecto:** Si se elimina un proyecto se eliminan todas las asignaciones de empleados asociadas a ese proyecto. Por lo tanto, al eliminar un proyecto, también se eliminarán automáticamente las asignaciones de empleados asociadas a ese proyecto.
- **Relación entre Perfil y Empleado:** Si se elimina un empleado todos los perfiles asociados a ese empleado. Por lo tanto, al eliminar un empleado, también se eliminarán automáticamente los perfiles asociados a ese empleado.

#### **RESTRICCIONES ON DELETE RESTRICT**

- **Relación entre Puesto y Departamento:** Si se intenta eliminar un departamento que tiene puestos asociados, se evita la eliminación para mantener la integridad de los datos.
- **Relación entre Empleado y Departamento:** Si se intenta eliminar un departamento que tiene empleados asociados, se evita la eliminación para mantener la integridad de los datos.

6. Investigue a qué se refieren las restricciones “ON DELETE SET NULL” Y “ON DELETE RESTRICT”. Implemente un ejemplo de cada una en `bd_practica.sql`.

- La restricción **ON DELETE SET NULL** permite que las tuplas de una tabla secundaria tengan el valor de su clave foránea como NULL cuando una tupla en la tabla principal sea eliminada.
- La restricción **ON DELETE RESTRICT** impide que una tupla de la tabla principal sea eliminada si existen tuplas relacionadas en la tabla secundaria, es decir, que si intentas eliminar una tupla en la tabla principal que está siendo referenciada por una o más tuplas en la tabla secundaria, la base de datos no permitirá la eliminación y arrojará un error.

7. Realice pruebas de borrado en cascada (ON DELETE CASCADE).

- a. Prueba de eliminación en cascada: Para este tipo de pruebas se inserta un conjunto de datos en las tablas relacionadas y, a continuación, elimina una fila en la tabla principal. Después hay que verificar que todas las filas relacionadas en la tabla secundaria se eliminen automáticamente y que no queden filas huérfanas.

**Para esta prueba intentamos eliminar el proyecto con id 60**

```
MariaDB [Edbm]> select * from proyecto;
```

id	descripcion	fecha_inicio	fecha_fin	presupuesto	departamento
60	Nulla tellus.	2020-08-03	2020-02-02	43907.8	4

```
60 rows in set (0.000 sec)
```

**Proyecto en la tabla principal**

```
MariaDB [Edbm]> select * from empleadoproyecto where proyecto=60;
```

id	empleado	proyecto
52	56	60

```
1 row in set (0.000 sec)
```

**Proyecto en la tabla secundaria**

```
MariaDB [Edbm]> delete from proyecto where id=60 and departamento=4;  
Query OK, 1 row affected (0.036 sec)
```

**Borramos el proyecto 60**

```
MariaDB [Edbm]> select * from proyecto where id=60;  
Empty set (0.000 sec)  
  
MariaDB [Edbm]> select * from empleadoproyecto where proyecto=60;  
Empty set (0.000 sec)
```

**Nos aseguramos que el proyecto se haya eliminado en la tabla principal y en la secundaria**

- b. Prueba de actualización en cascada: realizar una prueba similar a la anterior, pero en lugar de eliminar una fila en la tabla principal, actualiza la clave primaria de una fila. Asegúrate de que las claves foráneas en la tabla secundaria se actualicen correctamente.

**Para esta prueba usaremos el proyecto 59**

```
MariaDB [Edbm]> select * from proyecto where id=59;
```

id	descripcion	fecha_inicio	fecha_fin	presupuesto	departamento
59	Vestibulum ante ipsum primis in Duis faucibus accumsan odio.	2020-07-21	2023-10-05	54752.8	4

```
1 row in set (0.000 sec)  
  
MariaDB [Edbm]> select * from empleadoproyecto where proyecto=59;
```

id	empleado	proyecto
51	55	59

```
1 row in set (0.000 sec)
```

**Buscamos el proyecto en la tabla principal y en la secundaria**

```
MariaDB [Edbm]> update proyecto set id=100 where id=59;
Query OK, 1 row affected (0.037 sec)
Rows matched: 1  Changed: 1  Warnings: 0
```

**Actualizamos el id de la tabla principal de 59 a 100**

```
MariaDB [Edbm]> select * from proyecto where id=100;
+----+-----+-----+-----+-----+-----+
| id | descripcion | fecha_inicio | fecha_fin | presupuesto | departamento |
+----+-----+-----+-----+-----+-----+
| 100 | Vestibulum ante ipsum primis in Duis faucibus accumsan odio. | 2020-07-21 | 2023-10-05 | 54752.8 | 4 |
+----+-----+-----+-----+-----+-----+
1 row in set (0.000 sec)

MariaDB [Edbm]> select * from empleadoproyecto where proyecto=100;
+----+-----+-----+
| id | empleado | proyecto |
+----+-----+-----+
| 51 | 55 | 100 |
+----+-----+-----+
1 row in set (0.000 sec)
```

**Buscamos el proyecto con el id 100 en la tabla principal y en la secundaria para asegurarnos que se haya modificado en ambas**

8. Realice pruebas de borrado con restricción (ON DELETE RESTRICT).
  - a. Prueba de restricción de eliminación: Intenta eliminar una fila en la tabla principal que tenga filas relacionadas en la tabla secundaria. Verifica que la base de datos impida la eliminación y que muestre un error indicando que existe una restricción de integridad.

**Para esta prueba vamos a intentar eliminar un empleado que esté asignado a un proyecto**

```
MariaDB [Edbm]> select * from empleado where id=1;
+----+-----+-----+-----+-----+-----+-----+-----+-----+
| id | nombre | apellidos | fecha_nacimiento | direccion | fecha_alta | fecha_baja | salario | departamento | puesto |
+----+-----+-----+-----+-----+-----+-----+-----+-----+
| 1 | Hernando | Georgeou | 1981-10-07 | 353 Duke Alley | 2020-10-05 | 2020-09-25 | 67018.9 | 1 | 1 |
+----+-----+-----+-----+-----+-----+-----+-----+-----+
1 row in set (0.000 sec)
```

**Primero seleccionamos un empleado**

```
MariaDB [Edbm]> select * from empleadoproyecto where empleado=1;
+----+-----+-----+
| id | empleado | proyecto |
+----+-----+-----+
| 1 | 1 | 1 |
+----+-----+-----+
1 row in set (0.000 sec)
```

**Buscamos que tenga un proyecto asignado**

```
MariaDB [Edbm]> delete from empleado where id=1;
ERROR 1451 (23000): Cannot delete or update a parent row: a foreign key constraint fails (`edbm`.`empleadoproyecto`, CONSTRAINT `EmpleadoProyecto_fk0` FOREIGN KEY (`empleado`) REFERENCES `empleado` (`id`))
```

**Intentamos borrar el empleado y nos sale el siguiente error**

- b. Prueba de eliminación permitida: Intenta eliminar una fila en la tabla principal que no tenga filas relacionadas en la tabla secundaria. Verifica que la eliminación se realice correctamente sin ningún obstáculo.

**Para esto necesitamos eliminar un empleado que no esté asignado a un proyecto.**

```
MariaDB [Edbm]> select * from empleado where id=61;
+----+-----+-----+-----+-----+-----+-----+-----+-----+
| id | nombre | apellidos | fecha_nacimiento | direccion | fecha_alta | fecha_baja | salario | departamento | puesto |
+----+-----+-----+-----+-----+-----+-----+-----+-----+
| 61 | Jorge | Martinez | 1996-09-05 | Emiliano Zapata | 2023-10-05 | 2023-09-25 | 123456 | 1 | 1 |
+----+-----+-----+-----+-----+-----+-----+-----+-----+
1 row in set (0.000 sec)

MariaDB [Edbm]> select * from empleadoProyecto where empleado=61;
Empty set (0.000 sec)
```

**Usaremos el empleado con id 61 para esta prueba**

```
MariaDB [Edbm]> delete from empleado where id=61;
Query OK, 1 row affected (0.070 sec)
```

**Borramos el empleado**

```
MariaDB [Edbm]> select * from empleado where id=61;
Empty set (0.000 sec)
```

**Nos aseguramos que haya sido eliminado el empleado**

- c. Prueba de actualización restringida: Similar a la prueba de restricción de eliminación, intenta actualizar la clave primaria de una fila en la tabla principal cuando existen filas relacionadas en la tabla secundaria. Debería impedirse la actualización y mostrar un error.

**Para esta prueba usaremos al empleado con id=1**

```
MariaDB [Edbm]> select * from empleadoproyecto where empleado=1;
+----+-----+-----+
| id | empleado | proyecto |
+----+-----+-----+
| 1 | 1 | 1 |
+----+-----+-----+
1 row in set (0.000 sec)
```

**Nos aseguramos que el empleado con id 1 tenga un proyecto asignado**

```
MariaDB [Edbm]> update empleado set id=2 where id=1;
ERROR 1451 (23000): Cannot delete or update a parent row: a foreign key constraint fails (`edbm`.`empleadoproyecto`, CONSTRAINT `EmpleadoProyecto_fk0` FOREIGN KEY (`empleado`) REFERENCES `empleado` (`id`))
```

**Intentamos actualizar el id de la tabla principal**