

DISEÑO DE UNA BASE DE DATOS

28 DICIEMBRE

BASE DE DATOS

Cristian Martínez Roche

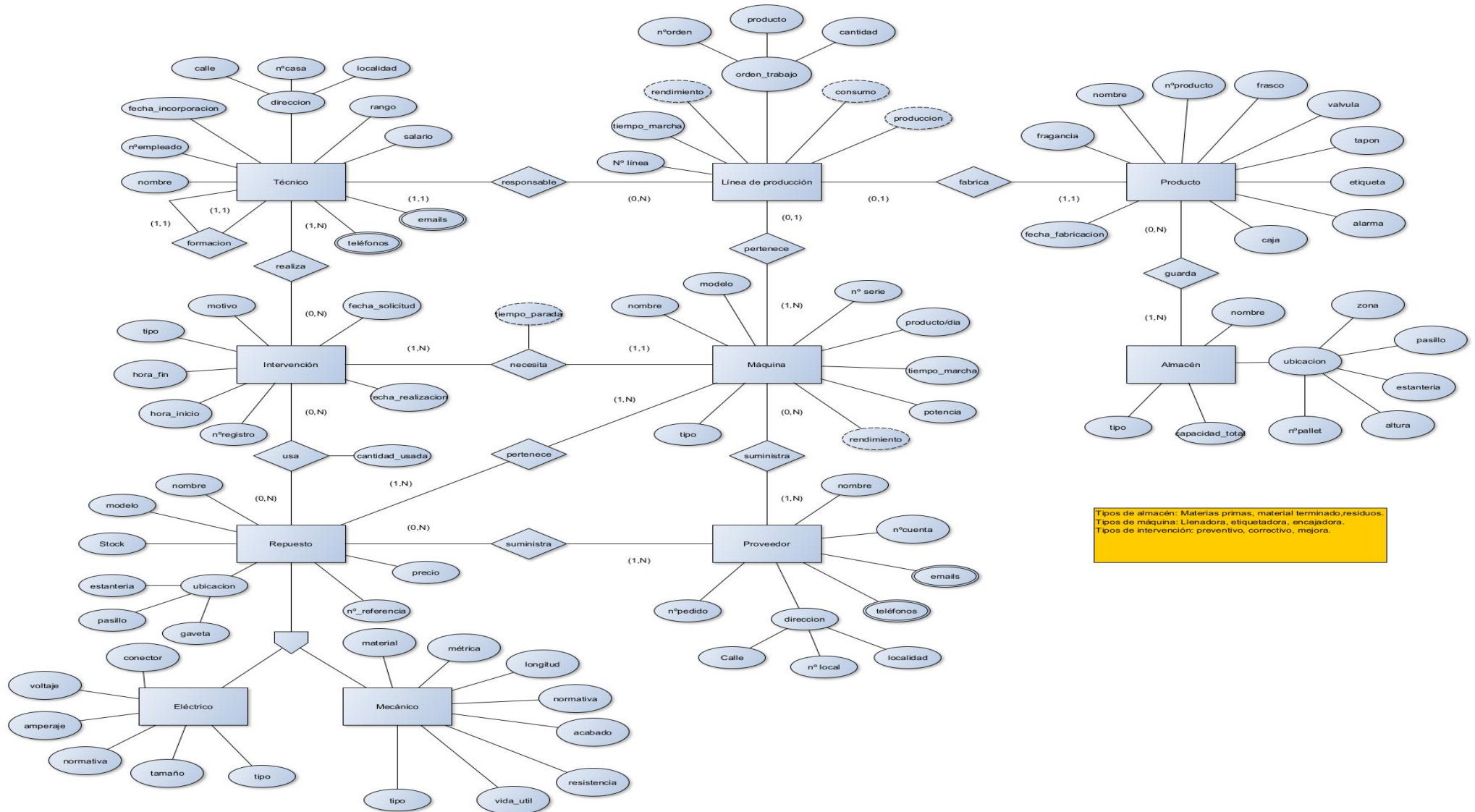


Enunciado:

Se desea diseñar una base de datos para gestionar la información de una fábrica de envasado de productos cosméticos. Los requisitos solicitados por el cliente son los siguientes:

- La fábrica cuenta con técnicos de los que interesa guardar su número de empleado, nombre, salario, fecha de incorporación, rango y su dirección (compuesta por calle, número y localidad). Dado que es importante localizarlos, se pueden registrar varios teléfonos y emails de contacto para un mismo técnico. Los técnicos veteranos forman a los técnicos noveles durante la jornada.
- La fábrica se organiza en líneas de producción. Cada línea puede estar formada por varias máquinas. Cada línea tiene que guardar los datos de su número de línea, tiempo en marcha, rendimiento diario, consumo total de la línea, producción y orden de trabajo (esta es compuesta por n° de orden, producto a realizar y cantidad). Cada línea tiene asignado un único técnico responsable, aunque un técnico puede ser responsable de varias líneas (o ninguna).
- Las máquinas dentro de la fábrica pueden ser parte de una sola línea de producción. De cada máquina interesa su número de serie, nombre, modelo, potencia, producto/día, tiempo de marcha y su eficiencia (rendimiento). Es importante clasificar la máquina según su tipo (llenadora, etiquetadora, encajadora).
- Los técnicos realizan intervenciones sobre las máquinas. Cada intervención tiene un número de registro, hora de inicio y fin, motivo, fecha de solicitud y de realización. Se debe clasificar el tipo de intervención (preventiva, correctiva, mejora). Una intervención puede realizarse por un técnico o varios. Se registrará el tiempo que la máquina está parada.
- Para las intervenciones se usan repuestos. De cada repuesto se guarda un código de referencia, nombre, modelo, precio, stock actual y su ubicación en el almacén (pasillo, gaveta, estantería). Se registra la cantidad de repuestos usada en cada intervención.
 - Los repuestos pueden ser de dos categorías:
 - Eléctricos: Se guardan los datos de voltaje, amperaje, normativa, tamaño y su tipo.
 - Mecánicos: Se guardan los datos de material, métrica, longitud, normativa, acabado, resistencia, vida útil y su tipo.
- Los repuestos y máquinas son suministrados por proveedores. De ellos se guarda el nombre, n° de cuenta bancaria y n° de pedido. Un proveedor puede tener varios teléfonos y varios emails. Para evitar falta de existencias los repuestos pueden ser suministrados por diferentes proveedores y los proveedores suministran diferentes repuestos, máquinas o ambos.
- Cada producto se fabrica en exclusiva en una línea de producción. De cada producto se guarda el nombre, n° de producto, frasco, válvula, tapón, etiqueta, caja, alarma, su fragancia y fecha de fabricación. Por motivos de producción puede haber productos que no se fabriquen en esos momentos.
- Los productos terminados se guardan en almacenes. De cada almacén interesa el nombre, su capacidad total y su tipo (materias primas, material terminado o residuos). Además, se detalla la ubicación precisa del producto (zona, pasillo, estantería, altura, n° de pallet).

Diagrama E/R:



Modelo relacional:

- **tecnico** (#id, nombre, n°empleado, fecha_incorporacion, calle, n°casa, localidad, rango, salario, -id_tecnico_formador)
- **emails_tecnico** (#id, -id_tecnico, email)
- **telefonos_tecnico** (#id, -id_tecnico, telefono)
- **intervencion** (#id, motivo, tipo, hora_inicio, hora_fin, n°registro, fecha_solicitud, fecha_realizacion, -id_maquina, tiempo_parada)
- **repuesto** (#id, nombre, modelo, stock, precio, n°referencia, estanteria, pasillo, gaveta)
- **electrico** (#id, conector, voltaje, amperaje, normativa, tamaño, tipo, -id_repuesto)
- **mecanico** (#id, material, metrica, longitud, normativa, acabado, resistencia, vida_util, tipo, -id_repuesto)
- **proveedor** (#id, nombre, n°cuenta, calle, n°local, localidad, n°pedido)
- **emails_proveedor** (#id, -id_proveedor, email)
- **telefonos_proveedor** (#id, -id_proveedor, telefono)
- **maquina** (#id, nombre, modelo, n°serie, producto/dia, tiempo_marcha, potencia, rendimiento, tipo, -id_linea_de_produccion)
- **linea_de_produccion** (#id, n°linea, tiempo_marcha, rendimiento, n°orden, producto, cantidad, consumo, produccion, -id_producto, -id_tecnico,)
- **producto** (#id, fragancia, nombre, n°producto, frasco, valvula, tapon, etiqueta, alarma, caja, fecha_fabricacion)
- **almacen** (#id, nombre, zona, pasillo, estanteria, altura, n°pallet, capacidad_total, tipo)
- **tecnico_realiza_intervencion** (#(-id_tecnico, -id_intervencion))
- **intervencion_usa_repuesto** (#(-id_intervencion, -id_repuesto), cantidad_usada)
- **proveedor_suministra_repuesto** (#(-id_proveedor, -id_repuesto))
- **proveedor_suministra_maquina** (#(-id_proveedor, -id_maquina))
- **repuesto_pertenece_maquina** (#(-id_repuesto, -id_maquina))
- **almacen_guarda_producto** (#(-id_almacen, -id_producto))

Script MySQL:

```
drop database if exists fabrica;  
create database fabrica;  
use fabrica;
```

```
create table tecnico (  
  id int unsigned primary key auto_increment,  
  nombre varchar (150) not null,  
  n°empleado int (10) unique not null,  
  fecha_incorporacion date not null,  
  calle varchar(100),  
  n°casa int(10),  
  localidad varchar(100),  
  rango varchar(25),  
  salario double (10,0),  
  id_tecnico_formador int unsigned,  
  foreign key(id_tecnico_formador) references tecnico(id)  
);
```

```
create table emails_tecnico (  
  id int unsigned primary key auto_increment,  
  id_tecnico int unsigned not null,  
  foreign key(id_tecnico) references tecnico(id),  
  email varchar (100)  
);
```

```
create table telefonos_tecnico (  
  id int unsigned primary key auto_increment,  
  id_tecnico int unsigned not null,  
  foreign key(id_tecnico) references tecnico(id),  
  telefono varchar(20)  
);
```

```
create table producto (  
  id int unsigned primary key auto_increment,  
  fragancia varchar (50) not null,  
  nombre varchar (20) not null,  
  n°producto varchar (50),  
  frasco varchar (50) not null,  
  tapon varchar (50) not null,  
  etiqueta varchar (50) not null,  
  alarma varchar (50) not null,  
  caja varchar (50) not null,  
  fecha_fabricacion date not null  
);
```

```

create table linea_de_produccion (
id int unsigned primary key auto_increment,
nºlinea varchar(20) not null,
rendimiento varchar(50),
nºorden varchar (50) not null,
producto varchar (50) not null,
cantidad int (10) not null,
tiempo_marcha time not null,
consumo varchar (10),
produccion int (10) not null,
id_producto int unsigned not null,
foreign key(id_producto) references producto(id),
id_tecnico int unsigned not null,
foreign key(id_tecnico) references tecnico(id)
);

```

```

create table maquina (
id int unsigned primary key auto_increment,
nombre varchar(20) not null,
tipo enum ('llenadora', 'etiquetadora', 'encajadora'),
modelo varchar(50) not null,
nºserie varchar(50),
producto_dia varchar(50) not null,
potencia varchar (20) not null,
tiempo_marcha time not null,
rendimiento varchar(20),
id_linea_de_produccion int unsigned not null,
foreign key(id_linea_de_produccion) references linea_de_produccion(id)
);

```

```

create table intervencion (
id int unsigned primary key auto_increment,
nºregistro varchar (20) not null,
motivo varchar (250) not null,
tipo enum ('preventivo', 'correctivo', 'mejora') not null,
fecha_solicitud date not null,
fecha_realizacion date not null,
hora_inicio time not null,
hora_fin time not null,
id_maquina int unsigned not null,
foreign key(id_maquina) references maquina(id),
tiempo_parada time
);

```

```
create table repuesto (  
id int unsigned primary key auto_increment,  
nºreferencia varchar (20) not null,  
nombre varchar (100) not null,  
modelo varchar (150) not null,  
stock int (10) not null,  
precio double (10,00) not null,  
estanteria varchar (10) not null,  
pasillo varchar (10) not null,  
gaveta varchar (50) not null  
);
```

```
create table electrico (  
id int unsigned primary key auto_increment,  
conector varchar (20) not null,  
voltaje varchar (10) not null,  
amperaje varchar (10) not null,  
tamaño varchar (10) not null,  
normativa varchar (150),  
tipo varchar (50) not null,  
id_repuesto int unsigned not null,  
foreign key(id_repuesto) references repuesto(id)  
);
```

```
create table mecanico (  
id int unsigned primary key auto_increment,  
material varchar (50) not null,  
metrica varchar (10) not null,  
longitud varchar (10) not null,  
acabado varchar (50),  
normativa varchar (150),  
resistencia varchar (50),  
vida_util varchar (50),  
tipo varchar (50) not null,  
id_repuesto int unsigned not null,  
foreign key(id_repuesto) references repuesto(id)  
);
```

```
create table proveedor (  
id int unsigned primary key auto_increment,  
nombre varchar(150) not null,  
nºcuenta varchar(30) unique not null,  
calle varchar(100),  
nºlocal int(10),  
localidad varchar(100),  
nºpedido varchar(25) not null  
);
```

```
create table emails_proveedor (  
id int unsigned primary key auto_increment,  
id_proveedor int unsigned not null,  
foreign key(id_proveedor) references proveedor(id),  
email varchar (100)  
);
```

```
create table telefonos_proveedor (  
id int unsigned primary key auto_increment,  
id_proveedor int unsigned not null,  
foreign key(id_proveedor) references proveedor(id),  
telefono varchar(20)  
);
```

```
create table almacen (  
id int unsigned primary key auto_increment,  
nombre varchar (100) not null,  
zona varchar (50) not null,  
pasillo varchar (10) not null,  
estanteria varchar (10) not null,  
altura varchar (10) not null,  
nºpallet int (10) not null,  
capacidad_total int (10) not null,  
tipo enum ('Materias_primas','Material_terminado', 'Residuos') not null  
);
```

```
create table tecnico_realiza_intervencion (  
id_tecnico int unsigned,  
foreign key(id_tecnico) references tecnico(id),  
id_intervencion int unsigned,  
foreign key(id_intervencion) references intervencion(id),  
primary key (id_tecnico, id_intervencion)  
);
```



```
create table intervencion_usa_repuesto (  
  id_intervencion int unsigned,  
  foreign key(id_intervencion) references intervencion(id),  
  id_repuesto int unsigned,  
  foreign key(id_repuesto) references repuesto(id),  
  primary key (id_intervencion, id_repuesto),  
  cantidad_usada int (5) unsigned not null  
);
```

```
create table proveedor_suministra_repuesto (  
  id_proveedor int unsigned,  
  foreign key(id_proveedor) references proveedor(id),  
  id_repuesto int unsigned,  
  foreign key(id_repuesto) references repuesto(id),  
  primary key (id_proveedor, id_repuesto)  
);
```

```
create table proveedor_suministra_maquina (  
  id_proveedor int unsigned,  
  foreign key(id_proveedor) references proveedor(id),  
  id_maquina int unsigned,  
  foreign key(id_maquina) references maquina(id),  
  primary key (id_proveedor, id_maquina)  
);
```

```
create table repuesto_pertenece_maquina (  
  id_repuesto int unsigned,  
  foreign key(id_repuesto) references repuesto(id),  
  id_maquina int unsigned,  
  foreign key(id_maquina) references maquina(id),  
  primary key (id_repuesto, id_maquina)  
);
```

```
create table almacen_guarda_producto (  
  id_almacen int unsigned,  
  foreign key(id_almacen) references almacen(id),  
  id_producto int unsigned,  
  foreign key(id_producto) references producto(id),  
  primary key (id_almacen, id_producto)  
);
```