

Universidad Nacional Autónoma de México

Facultad De Ciencias

Fundamentos de Bases de Datos
Proyecto Final

Nequiz Meza Antony Miguel

Peralta Luna Diego Alejandro

Romero Rodriguez Gabriel Eduardo

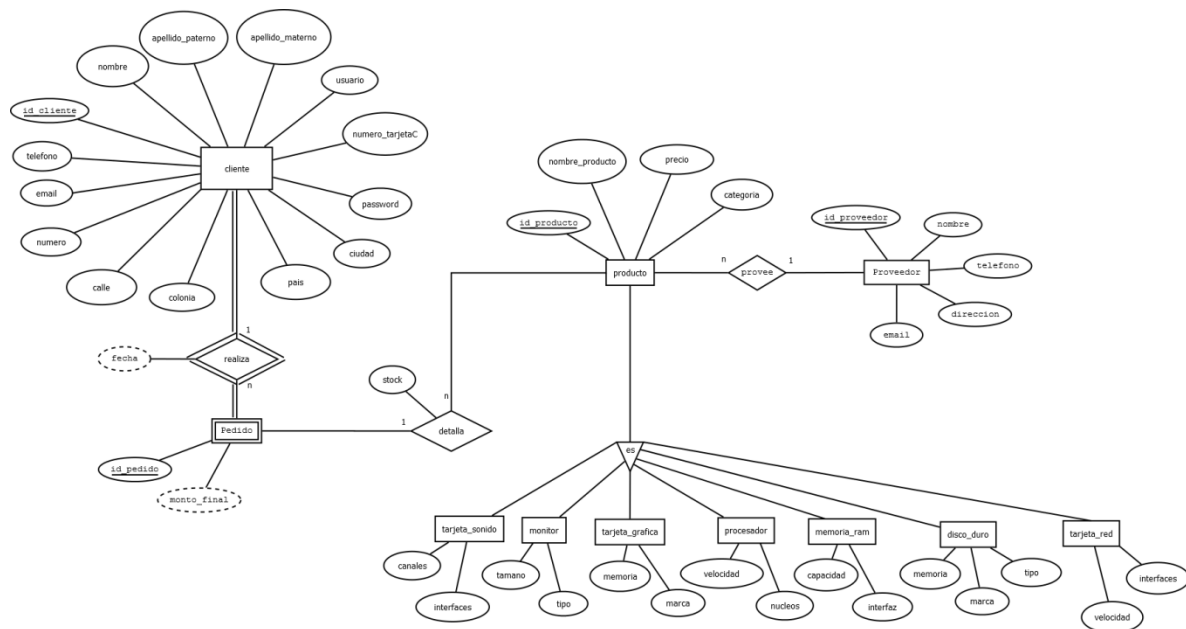
Vega Velazquez Alma Tania

Listado de Entregables 1 a 5

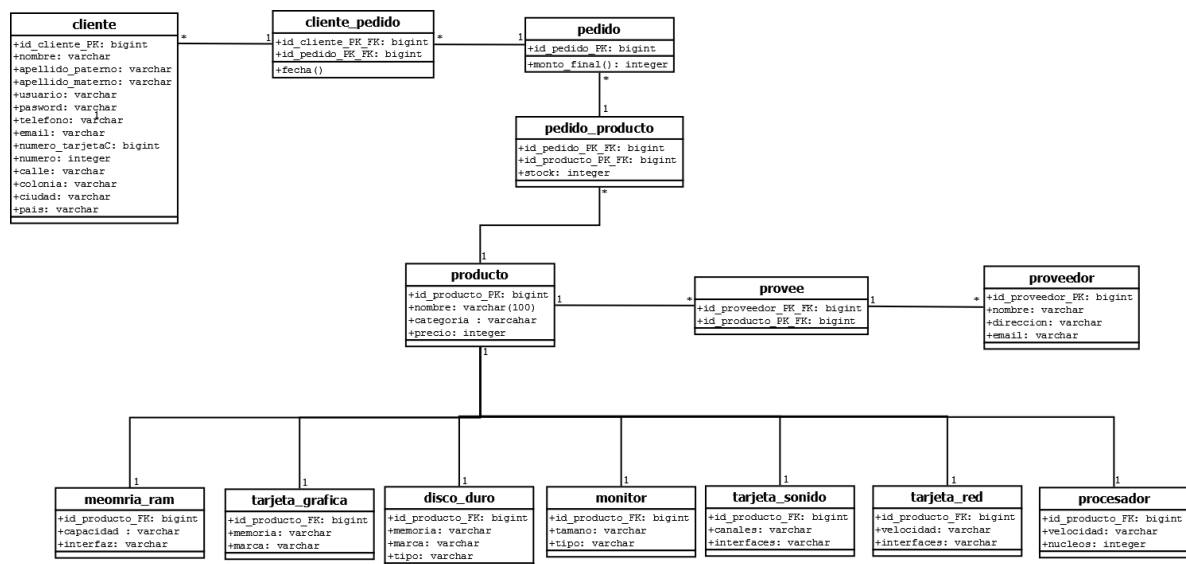
1. Listado de supuestos

1. El cliente solo cuenta con uno y solo un domicilio, un teléfono, un correo electrónico, un nombre de usuario, una contraseña y una cuenta de tarjeta de crédito.
2. El proveedor solo cuenta con uno y solo un domicilio y un correo electrónico.
3. Todos los productos del catálogo están disponibles para su venta. Existe al menos uno en el stock.
4. El proveedor siempre encuentra los productos que especifica cada orden.
5. El cliente siempre recibe sus productos.
6. El pago del cliente siempre es exitoso.
7. Todos los pedidos tienen al menos un producto.
8. Un pedido puede contener varios productos.
9. Cada producto tiene asignado un precio único.

2. Diagrama E-R



3. Diagrama de Clases



4. Listado de Dependencias Funcionales y Forma Normal de cada una de las tablas

1. Tabla cliente: Se encuentra en **3FN** pues existen DF transitivas. Además cumple las anteriores pues todo atributo no llave depende del atributo llave (2FN) y todo atributo es atómico (1FN).

Dependencias funcionales:

id_cliente -> numero, calle, colonia, ciudad, país, nombre_cliente, apellido_paterno_cliente, apellido_materno_cliente, email, teléfono, numero_tarjetaC, usuario, password

email -> teléfono

email -> usuario

email -> password

email -> numero_tarjetac

teléfono -> email

teléfono -> usuario

teléfono -> password

teléfono -> numero_tarjetac

usuario -> telefono

usuario -> email

usuario -> password

usuario -> numero_tarjetac

password -> usuario

password -> email

password -> teléfono

password -> numero_tarjetac

numero_tarjetac -> teléfono

numero_tarjetac -> email

numero_tarjetac -> usuario

numero_tarjetac -> password

2. Tabla cliente_pedido: Se encuentra en **FNBC** pues todo determinante es llave. Cumple las anteriores porque no existen DF transitivas (3FN), todo atributo no llave depende del atributo llave (2FN) y todo atributo es atómico (1FN).

Dependencia funcional:

(Id_cliente, id_pedido) -> fecha

3. Tabla disco_duro: Se encuentra en **FNBC** pues todo determinante es llave. Cumple las anteriores porque no existen DF transitivas (3FN), todo atributo no llave depende del atributo llave (2FN) y todo atributo es atómico (1FN).

Dependencia funcional:

Id_producto -> memoria, marca, tipo

4. Tabla memoria_ram: Se encuentra en **FNBC** pues todo determinante es llave. Cumple las anteriores porque no existen DF transitivas (3FN), todo atributo no llave depende del atributo llave (2FN) y todo atributo es atómico (1FN).

Dependencia funcional:

Id_producto -> capacidad, interfaz

5. Tabla monitor: Se encuentra en **FNBC** pues todo determinante es llave. Cumple las anteriores porque no existen DF transitivas (3FN), todo atributo no llave depende del atributo llave (2FN) y todo atributo es atómico (1FN).

Dependencia funcional:

Id_producto -> tamaño, tipo

6. Tabla pedido: Se encuentra en **FNBC** pues todo determinante es llave. Cumple las anteriores porque no existen DF transitivas (3FN), todo atributo no llave depende del atributo llave (2FN) y todo atributo es atómico (1FN).

Dependencia funcional:

Id_pedido -> monto_final

7. Tabla pedido_producto: Se encuentra en **FNBC** pues todo determinante es llave. Cumple las anteriores porque no existen DF transitivas (3FN), todo atributo no llave depende del atributo llave (2FN) y todo atributo es atómico (1FN).

Dependencia funcional:

Id_pedido, id_producto -> stock

8. Tabla procesador: Se encuentra en **FNBC** pues todo determinante es llave. Cumple las anteriores porque no existen DF transitivas (3FN), todo atributo no llave depende del atributo llave (2FN) y todo atributo es atómico (1FN).

Dependencia funcional:

Id_producto -> velocidad, nucleos

9. Tabla producto: Se encuentra en **FNBC** pues todo determinante es llave. Cumple las anteriores porque no existen DF transitivas (3FN), todo atributo no llave depende del atributo llave (2FN) y todo atributo es atómico (1FN).

Dependencia funcional:

Id_producto -> nombre, precio, categoría

10. Tabla proveedor: Se encuentra en **3FN** pues existen DF transitivas. Además cumple las anteriores pues todo atributo no llave depende del atributo llave (2FN) y todo atributo es atómico (1FN). Aunque dirección contenga todos los datos de la dirección en un solo varchar, es un atributo atómico pues solo está permitida una sola dirección por proveedor (no es multivaluado).

Dependencias funcionales:

Id_proveedor -> nombre, direccion, email

direccion -> email

direccion -> nombre

email -> direccion

email -> nombre

nombre -> direccion

nombre -> email

11. Tabla tarjeta_grafica: Se encuentra en **FNBC** pues todo determinante es llave. Cumple las anteriores porque no existen DF transitivas (3FN), todo atributo no llave depende del atributo llave (2FN) y todo atributo es atómico (1FN).

Dependencia funcional:

Id_producto -> memoria, marca

12. Tabla tarjeta_red: Se encuentra en **FNBC** pues todo determinante es llave. Cumple las anteriores porque no existen DF transitivas (3FN), todo atributo no llave depende del atributo llave (2FN) y todo atributo es atómico (1FN).

Dependencia funcional:

Id_producto -> interfaces, velocidad

13. Tabla tarjeta_sonido: Se encuentra en **FNBC** pues todo determinante es llave. Cumple las anteriores porque no existen DF transitivas (3FN), todo atributo no llave depende del atributo llave (2FN) y todo atributo es atómico (1FN).

Dependencia funcional:

Id_producto -> canales, interfaces

14. Tabla provee: Tabla tarjeta_sonido: Se encuentra en **FNBC** pues todo determinante es llave. Cumple las anteriores porque no existen DF transitivas (3FN), todo atributo no llave depende del atributo llave (2FN) y todo atributo es atómico (1FN).

Llave: Id_producto, id_proveedor

5. Listado de las restricciones de integridad Implementadas

Las restricciones de integridad fueron implementadas con llaves primarias (PRIMARY KEY), llaves foráneas (FOREIGN KEY), CHECKS y NOT NULLs.

Tabla cliente:

```
id_cliente bigint NOT NULL,  
nombre character varying NOT NULL,  
apellido_paterno character varying NOT NULL,  
apellido_materno character varying NOT NULL,  
usuario character varying NOT NULL,  
password character varying NOT NULL,  
telefono character varying NOT NULL,  
email character varying NOT NULL,  
numero_tarjetac bigint,  
calle character varying,  
colonia character varying,  
ciudad character varying ,  
pais character varyings,  
CONSTRAINT cliente_pkey PRIMARY KEY (id_cliente),  
CONSTRAINT cliente_email_check CHECK (email::text ~* '^[A-Za-z0-9._%~]+@[A-Za-z0-9.-]+[.][A-Za-zst]+$',::text),  
CONSTRAINT cliente_id_cliente_check CHECK (id_cliente > 0),  
CONSTRAINT cliente_numero_tarjetac_check CHECK (numero_tarjetac > 0)
```

Tabla cliente_pedido:

```
CONSTRAINT cliente_pedido_pkey PRIMARY KEY (id_cliente, id_pedido),  
CONSTRAINT cliente_pedido_id_cliente_fkey FOREIGN KEY (id_cliente)  
REFERENCES cliente (id_cliente) MATCH SIMPLE  
ON UPDATE CASCADE ON DELETE CASCADE,  
CONSTRAINT cliente_pedido_id_pedido_fkey FOREIGN KEY (id_pedido)  
REFERENCES pedido (id_pedido) MATCH SIMPLE  
ON UPDATE NO ACTION ON DELETE NO ACTION,  
CONSTRAINT cliente_pedido_id_cliente_key UNIQUE (id_cliente),  
CONSTRAINT cliente_pedido_id_pedido_key UNIQUE (id_pedido)
```

Tabla disco_duro:

```
id_producto bigint NOT NULL,  
CONSTRAINT disco_duro_pkey PRIMARY KEY (id_producto),  
CONSTRAINT disco_duro_id_producto_fkey FOREIGN KEY (id_producto)  
REFERENCES producto (id_producto) MATCH SIMPLE  
ON UPDATE NO ACTION ON DELETE NO ACTION
```

Tabla memoria_ram:

```
id_producto bigint NOT NULL,  
CONSTRAINT memoria_ram_pkey PRIMARY KEY (id_producto),  
CONSTRAINT memoria_ram_id_producto_fkey FOREIGN KEY (id_producto)
```

```
REFERENCES producto (id_producto) MATCH SIMPLE
ON UPDATE NO ACTION ON DELETE NO ACTION
```

Tabla monitor:

```
id_producto bigint NOT NULL,
CONSTRAINT monitor_pkey PRIMARY KEY (id_producto),
CONSTRAINT monitor_id_producto_fkey FOREIGN KEY (id_producto)
REFERENCES producto (id_producto) MATCH SIMPLE
ON UPDATE NO ACTION ON DELETE NO ACTION
```

Tabla pedido:

```
id_pedido bigint NOT NULL,
CONSTRAINT pedido_pkey PRIMARY KEY (id_pedido),
CONSTRAINT pedido_id_pedido_check CHECK (id_pedido > 0),
CONSTRAINT pedido_monto_final_check CHECK (monto_final >= 0::double precision)
```

Tabla procesador:

```
id_producto bigint NOT NULL,
velocidad character varying,
nucleos integer,
CONSTRAINT procesador_pkey PRIMARY KEY (id_producto),
CONSTRAINT procesador_id_producto_fkey FOREIGN KEY (id_producto)
REFERENCES producto (id_producto) MATCH SIMPLE
ON UPDATE NO ACTION ON DELETE NO ACTION,
CONSTRAINT procesador_nucleos_check CHECK (nucleos > 0)
```

Tabla producto:

```
id_producto bigint NOT NULL,
precio integer NOT NULL,
CONSTRAINT producto_pkey PRIMARY KEY (id_producto),
CONSTRAINT producto_id_producto_check CHECK (id_producto > 0),
CONSTRAINT producto_precio_check CHECK (precio > 0)
```

Tabla producto_proveedor:

```
id_producto serial NOT NULL,
id_proveedor serial NOT NULL,
CONSTRAINT producto_proveedor_pkey PRIMARY KEY (id_producto, id_proveedor),
CONSTRAINT producto_proveedor_id_producto_fkey FOREIGN KEY (id_producto)
REFERENCES producto (id_producto) MATCH SIMPLE
ON UPDATE NO ACTION ON DELETE NO ACTION,
CONSTRAINT producto_proveedor_id_proveedor_fkey FOREIGN KEY (id_proveedor)
REFERENCES proveedor (id_proveedor) MATCH SIMPLE
ON UPDATE NO ACTION ON DELETE NO ACTION,
CONSTRAINT producto_proveedor_id_producto_key UNIQUE (id_producto),
CONSTRAINT producto_proveedor_id_proveedor_key UNIQUE (id_proveedor)
```


Tabla proveedor:

```
id_proveedor bigint NOT NULL,  
CONSTRAINT proveedor_pkey PRIMARY KEY (id_proveedor),  
CONSTRAINT proveedor_email_check CHECK (email::text ~* '^[A-Za-z0-9._%~]+@[A-  
Za-z0-9.-]+[.][A-Za-z]+$'::text),  
CONSTRAINT proveedor_id_proveedor_check CHECK (id_proveedor > 0)
```

Tabla tarjeta_grafica:

```
id_producto bigint NOT NULL,  
CONSTRAINT tarjeta_grafica_pkey PRIMARY KEY (id_producto),  
CONSTRAINT tarjeta_grafica_id_producto_fkey FOREIGN KEY (id_producto)  
REFERENCES producto (id_producto) MATCH SIMPLE  
ON UPDATE NO ACTION ON DELETE NO ACTION
```

Tabla tarjeta_red:

```
id_producto bigint NOT NULL,  
CONSTRAINT tarjeta_red_pkey PRIMARY KEY (id_producto),  
CONSTRAINT tarjeta_red_id_producto_fkey FOREIGN KEY (id_producto)  
REFERENCES producto (id_producto) MATCH SIMPLE  
ON UPDATE NO ACTION ON DELETE NO ACTION
```

Tabla tarjeta_sonido:

```
id_producto bigint NOT NULL,  
CONSTRAINT tarjeta_sonido_pkey PRIMARY KEY (id_producto),  
CONSTRAINT tarjeta_sonido_id_producto_fkey FOREIGN KEY (id_producto)  
REFERENCES producto (id_producto) MATCH SIMPLE  
ON UPDATE NO ACTION ON DELETE NO ACTION
```

Tabla provee:

```
id_proveedor bigint NOT NULL,  
id_producto bigint NOT NULL,  
CONSTRAINT provee_pkey PRIMARY KEY (id_producto, id_proveedor),  
CONSTRAINT provee_id_producto_fkey FOREIGN KEY (id_producto)  
REFERENCES producto (id_producto) MATCH SIMPLE  
ON UPDATE NO ACTION ON DELETE NO ACTION,  
CONSTRAINT provee_id_proveedor_fkey FOREIGN KEY (id_proveedor)  
REFERENCES proveedor (id_proveedor) MATCH SIMPLE  
ON UPDATE NO ACTION ON DELETE NO ACTION,  
CONSTRAINT provee_id_producto_key UNIQUE (id_producto),  
CONSTRAINT provee_id_proveedor_key UNIQUE (id_proveedor)
```