

Proyecto Final Tienda Online de Productos Electrónicos

Alumno: **Aaron Rodrigo Ramos Reyes**

Profesor: **Guillermo Monroy Rodríguez**

Materia: **Bases de Datos**

Universidad Autónoma Metropolitana

Unidad Cuajimalpa

30 de julio de 2025

Índice

| | |
|---|----------|
| 1. Introducción | 2 |
| 1.1. Introducción al Proyecto | 2 |
| 2. Diagrama ER y justificación de la normalización | 2 |
| 2.1. Diagrama Entidad-Relación | 2 |
| 2.2. Justificación de la normalización | 3 |
| 3. Implementación en MySQL | 4 |
| 4. Implementación en SQL | 4 |
| 4.1. Creación de la base de datos y tablas | 4 |
| 4.2. Tabla Empleados | 4 |
| 4.3. Tabla Tabulador | 4 |

1 Introducción

-

1.1 Introducción al Proyecto

El proyecto consiste en el diseño e implementación de una base de datos en **MySQL**, la cual debe cumplir con todos los requisitos descritos en el documento "*ProyectoFinalBD*".

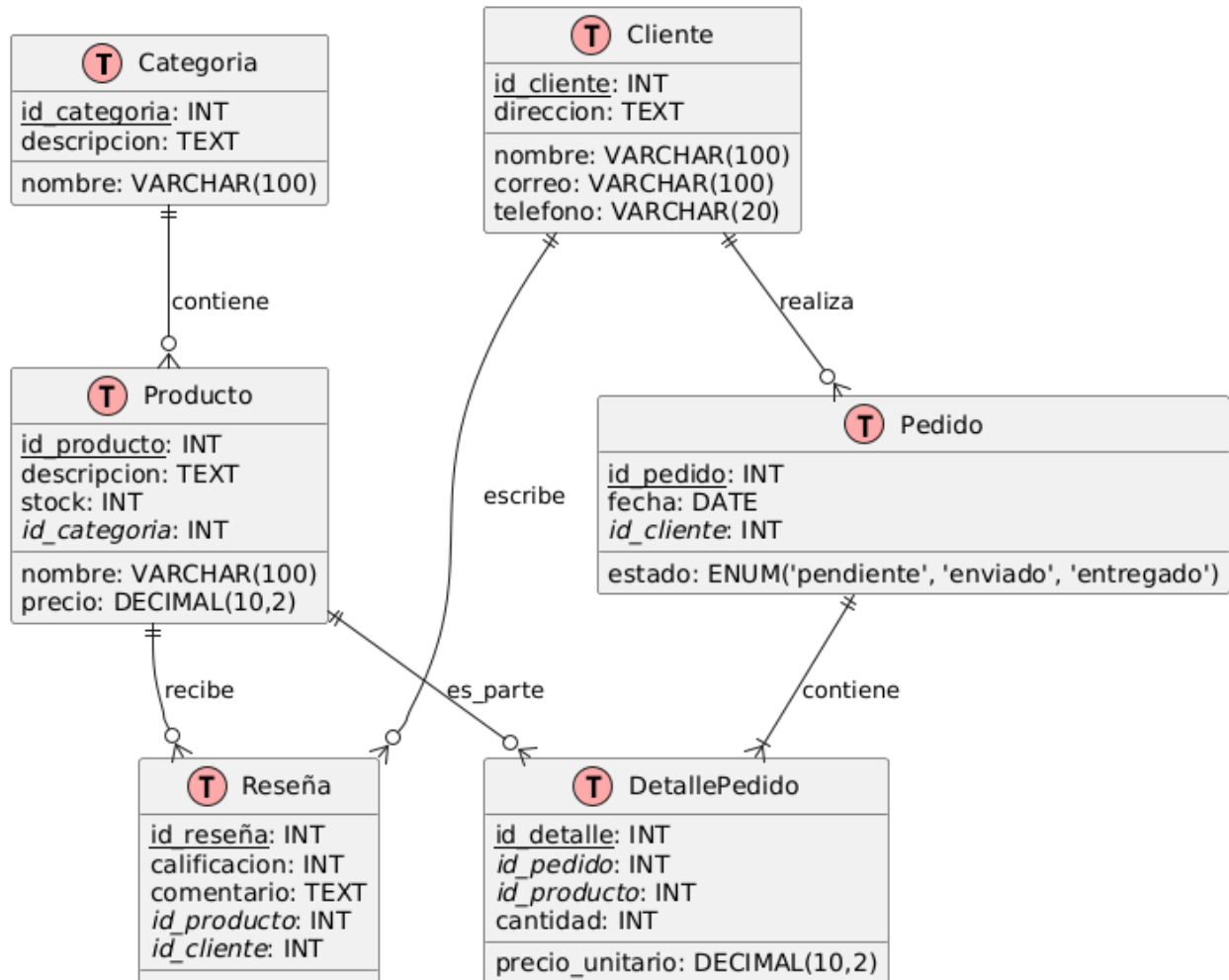
Con este propósito, se requerirá el uso y reforzamiento de tecnologías como **PlantUML** para la elaboración de diagramas, **L^AT_EX** para generar documentación clara, estructurada y modificable, y **MySQL** para la implementación de la base de datos. Asimismo, se utilizarán otras herramientas como los *stored procedures* para extender la funcionalidad de la base de datos.

Adicionalmente, se plantea la posibilidad de desarrollar un sistema en **Java** utilizando una arquitectura de tres capas, lo cual será opcional y dependerá del tiempo disponible para la entrega. Esta etapa adicional busca integrar y aplicar conocimientos adquiridos en otros cursos del plan de estudios.

2 Diagrama ER y justificación de la normalización

2.1 Diagrama Entidad-Relación

El siguiente diagrama muestra el modelo entidad-relación para el sistema de gestión de una tienda en línea de productos electrónicos. Se incluyen entidades como productos, clientes, pedidos, categorías, y reseñas, así como las relaciones entre ellas.



2.2 Justificación de la normalización

El modelo ha sido diseñado cumpliendo con las tres primeras formas normales para asegurar consistencia, evitar redundancia y facilitar el mantenimiento de los datos:

- **Primera Forma Normal (1FN):** Todas las columnas contienen valores atómicos y no hay grupos repetitivos.
- **Segunda Forma Normal (2FN):** No existen dependencias parciales; todos los atributos no clave dependen completamente de la clave primaria.
- **Tercera Forma Normal (3FN):** No hay dependencias transitivas; cada atributo no clave depende solamente de la clave primaria.

3 Implementación en MySQL

4 Implementación en SQL

4.1 Creación de la base de datos y tablas

```
CREATE DATABASE EMPRESA;  
USE EMPRESA;
```

4.2 Tabla Empleados

```
CREATE TABLE Empleados (  
    id INT AUTO_INCREMENT PRIMARY KEY,  
    nombre VARCHAR(100),  
    fecha_de_ingreso DATE,  
    salario DECIMAL(10,2),  
    categoria VARCHAR(100)  
);
```

Descripción de campos:

- id: Identificador único del empleado (clave primaria)
- nombre: Nombre completo del empleado
- fecha_de_ingreso: Fecha de contratación
- salario: Remuneración del empleado
- categoria: Categoría o puesto del empleado

4.3 Tabla Tabulador

```
CREATE TABLE Tabulador (  
    id INT AUTO_INCREMENT PRIMARY KEY,  
    nombre_puesto VARCHAR(100),  
    sueldo DECIMAL(10,2),  
    dias_antiguedad INT  
);
```

Descripción de campos:

- `id`: Identificador único del registro (clave primaria)
- `nombre_puesto`: Nombre del puesto o categoría
- `sueldo`: Salario base para el puesto
- `dias_antigüedad`: Días requeridos para antigüedad en el puesto