

► UMG Ingeniería en Sistemas

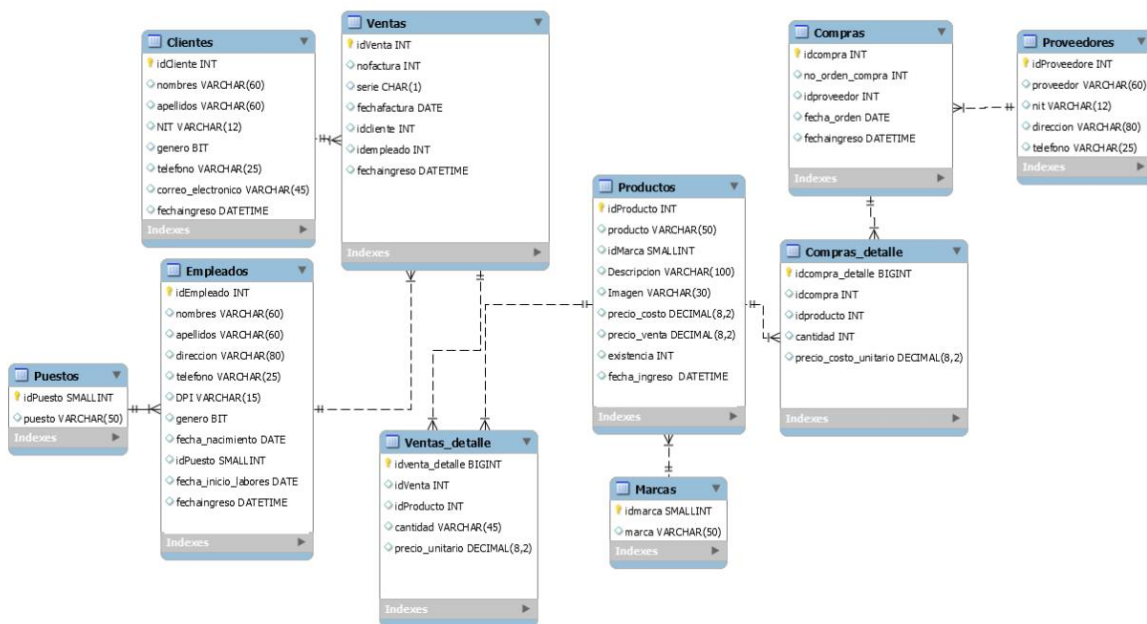


Objetivos:

Demostrar las habilidades prácticas de programación en Java así como la habilidad del diseño y creación de sitios web en JSP.

Fase I: (20 pts)

- I) Crear una base de datos en **mysql**.



- 2) Crear el mantenimiento web (CRUD) de la tabla **Puestos**.
- 3) Crear el mantenimiento web (CRUD) de la tabla **Empleados** la cual deberá de mostrar un combo con los **puestos** de la tabla **Puestos** y un hipervínculo que redirecciones al mantenimiento de **Puestos** y viceversa.
- 4) Crear el mantenimiento web (CRUD) de la tabla **Clientes**.
- 5) Crear el mantenimiento web (CRUD) de la tabla **Proveedores**.
- 6) Crear el mantenimiento web (CRUD) de la tabla **Marcas**.
- 7) Crear el mantenimiento web (CRUD) de la tabla **Productos** el cual deberá de mostrar un combo con las **marcas** de la tabla **Marcas** y un hipervínculo que redirecciones al mantenimiento de **Marcas** y viceversa. Este mantenimiento deberá de permitir Guardar una **IMAGEN** del producto en el servidor,(no en la base de datos ahí solo deberá de estar la

URL de la IMAGEN) y cuando se realice una búsqueda del producto esta deberá demostrar la imagen almacenada.

- 8) Crear un mantenimiento web (CRUD) de tipo **MAESTRO DETALLE** de las tablas **Ventas y Ventas_Detalle**, es decir en un solo mantenimiento se deberá de guardar en las dos tablas. El mantenimiento deberá de mostrar un combo con los nombres y nit de los **clientes** de la tabla **Clientes** y un hipervínculo que redirecciones al mantenimiento de **Clientes** y viceversa. El mantenimiento deberá de mostrar un combo con los nombres de los **empleados** de la tabla **Empleados** y un hipervínculo que redirecciones al mantenimiento de **Empleados** y viceversa. Cuando se ingrese una venta el saldo del producto de la tabla **Producto** deberá de disminuir.
- 9) Crear un mantenimiento web (CRUD) de tipo **MAESTRO DETALLE** de las tablas **Compras y Compras_Detalle**, es decir en un solo mantenimiento se deberá de guardar en las dos tablas. El mantenimiento deberá de mostrar un combo con los nombres de los **proveedores** de la tabla **Proveedores** y un hipervínculo que redirecciones al mantenimiento de **Proveedores** y viceversa. Cuando se ingrese una compra el saldo del producto de la tabla **Producto** deberá de aumentar y el precio_costo deberá de actualizarse, así como el precio_venta pero este con un 25% más del precio_costo
- 10) Deberá de crear un login para ingresar a la aplicación (Crear una tabla en la base de datos usuarios para almacenar el usuario y contraseña)
- 11) Deberá de crear un menú principal **DINAMICO** (Crear una tabla en la base de Datos para los menús) por medio de Arboles con la siguiente estructura.
 1. Productos
 - 1.1. Marcas
 2. Ventas
 - 2.1. Clientes
 - 2.2. Empleados
 - 2.2.1. Puestos
 3. Compras
 - 3.1. Proveedores
 4. Reportes
- 12) Todos los mantenimientos deberán de llevar las validaciones básicas para el ingreso de datos.
- 13) Crear como mínimo 5 reportes básicos con JasperReports u otra alternativa a su elección.

Fase II (5 pts)

Desarrollo de Aplicación Móvil

- 1) Crear una aplicación móvil Nativa en **Android** con Android Studio utilizando Java, la aplicación deberá de permitir visualizar una lista todos los productos de la Tabla Productos (incluido imagen), consumiendo un **API REST**.
 - a. La aplicación deberá de permitir visualizar una lista con el nombre, precio, marca, descripción e imagen del producto.
 - b. Al presionar o seleccionar el producto este deberá de mostrara información detallada del producto.
 - c. La aplicación deberá poseer filtros que faciliten la búsqueda de búsqueda.

Requisitos

- 1) Grupos máximo de 5 personas
- 2) Aplicar los conceptos de la Programación Orientada a Objetos (POO).
- 3) Toda la aplicación deberá de ser en Java Web.
- 4) Utilizar el patrón de arquitectura de software MVC (Modelo–vista–controlador)
- 5) Deberá de funcionar en red, es decir deberán de publicar la aplicación en un servidor (local), y las maquinas cliente deberán de acceder a ella por medio de una navegador web.
- 6) La aplicación deberá de ser amigable e interactiva para el usuario (usar bootstrap como recomendación no como obligación).
- 7) Deberá de utilizar Git como controlador de versiones.
- 8) Deberá de utilizar GitHub para alojar su proyecto.
- 9) Deberá de utilizar <https://dev.azure.com/> para darle seguimiento al proyecto y aplicar una metodología de desarrollo ágil.
- 10) Deberán de Realizar un video y presentación final del proyecto.