

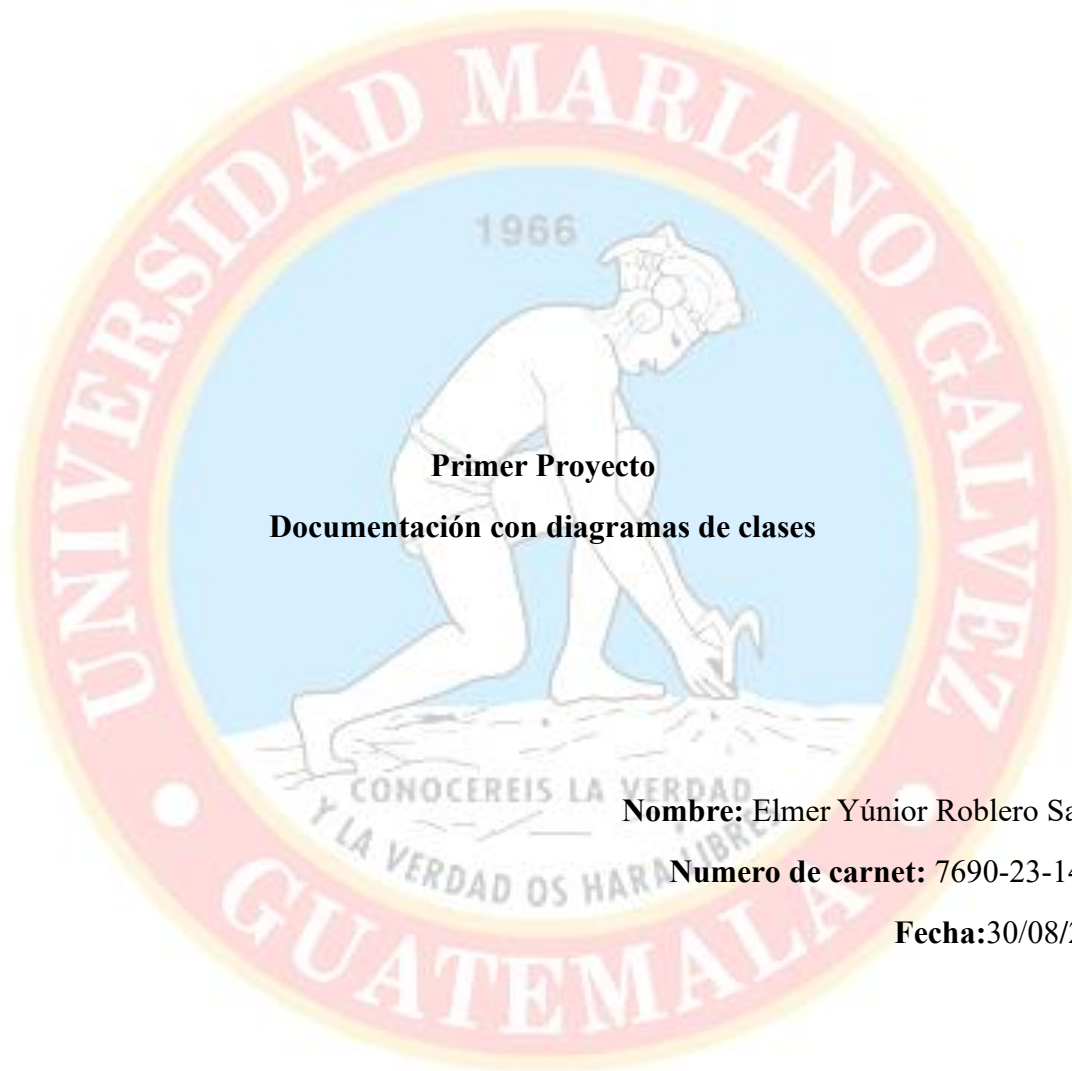
# Universidad Mariano Gálvez

**Centro Regional:** Boca del Monte

**Facultad:** Ingeniería en sistemas

**Curso:** PROGRAMACIÓN II

**Sección:** A



**Primer Proyecto**

**Documentación con diagramas de clases**

**Nombre:** Elmer Yúnior Roblero Santos

**Numero de carnet:** 7690-23-14847

**Fecha:** 30/08/2024

**Guatemala, de 2024**

## Documentación del Proyecto

### Descripción General

Este proyecto es una aplicación de gestión para una empresa ficticia llamada "Los Globitos". La aplicación permite gestionar productos, clientes, vendedores, proveedores, ventas, compras e importaciones a través de un menú interactivo.

### Estructura del Proyecto

#### 1. Main.java

La clase principal que contiene el método main, el punto de entrada de la aplicación.

- **Método main(String[] args):**
  - **Descripción:** Inicializa la empresa y el menú, y lanza la interfaz de usuario.
  - **Parámetros:**
    - args: Argumentos de línea de comandos.

#### 2. Empresa.java

Gestiona la información de la empresa y las entidades relacionadas (productos, clientes, vendedores, etc.).

- **Atributos:**
  - nombre: String - Nombre de la empresa.
  - productos: ArrayList<Producto> - Lista de productos de la empresa.
  - clientes: ArrayList<Cliente> - Lista de clientes.
  - vendedores: ArrayList<Vendedor> - Lista de vendedores.
  - proveedores: ArrayList<Proveedor> - Lista de proveedores.
  - facturas: ArrayList<Factura> - Lista de facturas.
  - logistica: ArrayList<Logistica> - Lista de operaciones logísticas.
  - categorias: ArrayList<Categoria> - Lista de categorías de productos.
- **Métodos:**
  - Métodos para agregar, modificar y mostrar productos, clientes, vendedores, etc.

#### 3. Menu.java

Gestiona el menú interactivo de la aplicación.

- **Atributos:**

- empresa: Empresa - Instancia de la clase Empresa para acceder a la información de la empresa.
- scanner: Scanner - Para leer la entrada del usuario.
- **Métodos:**
  - mostrar(): Muestra el menú principal y gestiona la selección de opciones.
  - procesarOpcion(int opcion): Procesa la opción seleccionada por el usuario.

#### 4. Producto.java

Representa un producto con sus atributos y métodos relacionados.

- **Atributos:**
  - codigo: String - Código del producto.
  - nombre: String - Nombre del producto.
  - cantidad: int - Cantidad disponible del producto.
- **Métodos:**
  - getCodigo(), setCodigo(String codigo): Métodos para obtener y establecer el código del producto.
  - getNombre(), setNombre(String nombre): Métodos para obtener y establecer el nombre del producto.
  - getCantidad(), setCantidad(int cantidad): Métodos para obtener y establecer la cantidad del producto.
  - toString(): Representación en cadena del producto.

#### 5. Venta.java

Representa una venta realizada en la empresa.

- **Atributos:**
  - producto: Producto - Producto vendido.
  - cliente: Cliente - Cliente que realizó la compra.
  - cantidadVendida: int - Cantidad de producto vendida.
- **Métodos:**
  - Métodos relevantes para gestionar ventas.

#### 6. Compra.java

Representa una compra de productos a proveedores.

- **Atributos:**

- producto: Producto - Producto comprado.
- proveedor: Proveedor - Proveedor del producto.
- cantidadComprada: int - Cantidad de producto comprada.

- **Métodos:**

- Métodos relevantes para gestionar compras.

## 7. Importacion.java

Representa la importación de productos desde otros países.

- **Atributos:**

- producto: Producto - Producto importado.
- paisOrigen: String - País de origen del producto.
- detallesEnvio: String - Detalles del envío.

- **Métodos:**

- Métodos relevantes para gestionar importaciones.

## 8. Cliente.java

Representa un cliente de la empresa.

- **Atributos:**

- codigo: String - Código del cliente.
- nombre: String - Nombre del cliente.

- **Métodos:**

- getCodigo(), setCodigo(String codigo): Métodos para obtener y establecer el código del cliente.
- getNombre(), setNombre(String nombre): Métodos para obtener y establecer el nombre del cliente.

## 9. Proveedor.java

Representa un proveedor de la empresa.

- **Atributos:**

- nombre: String - Nombre del proveedor.
- nombreEmpresa: String - Nombre de la empresa del proveedor.



- producto: String - Producto que el proveedor suministra.
- **Métodos:**
  - getNombre(), setNombre(String nombre): Métodos para obtener y establecer el nombre del proveedor.
  - getNombreEmpresa(), setNombreEmpresa(String nombreEmpresa): Métodos para obtener y establecer el nombre de la empresa del proveedor.
  - getProducto(), setProducto(String producto): Métodos para obtener y establecer el producto suministrado por el proveedor.

## 10. Vendedor.java

Representa un vendedor en la empresa.

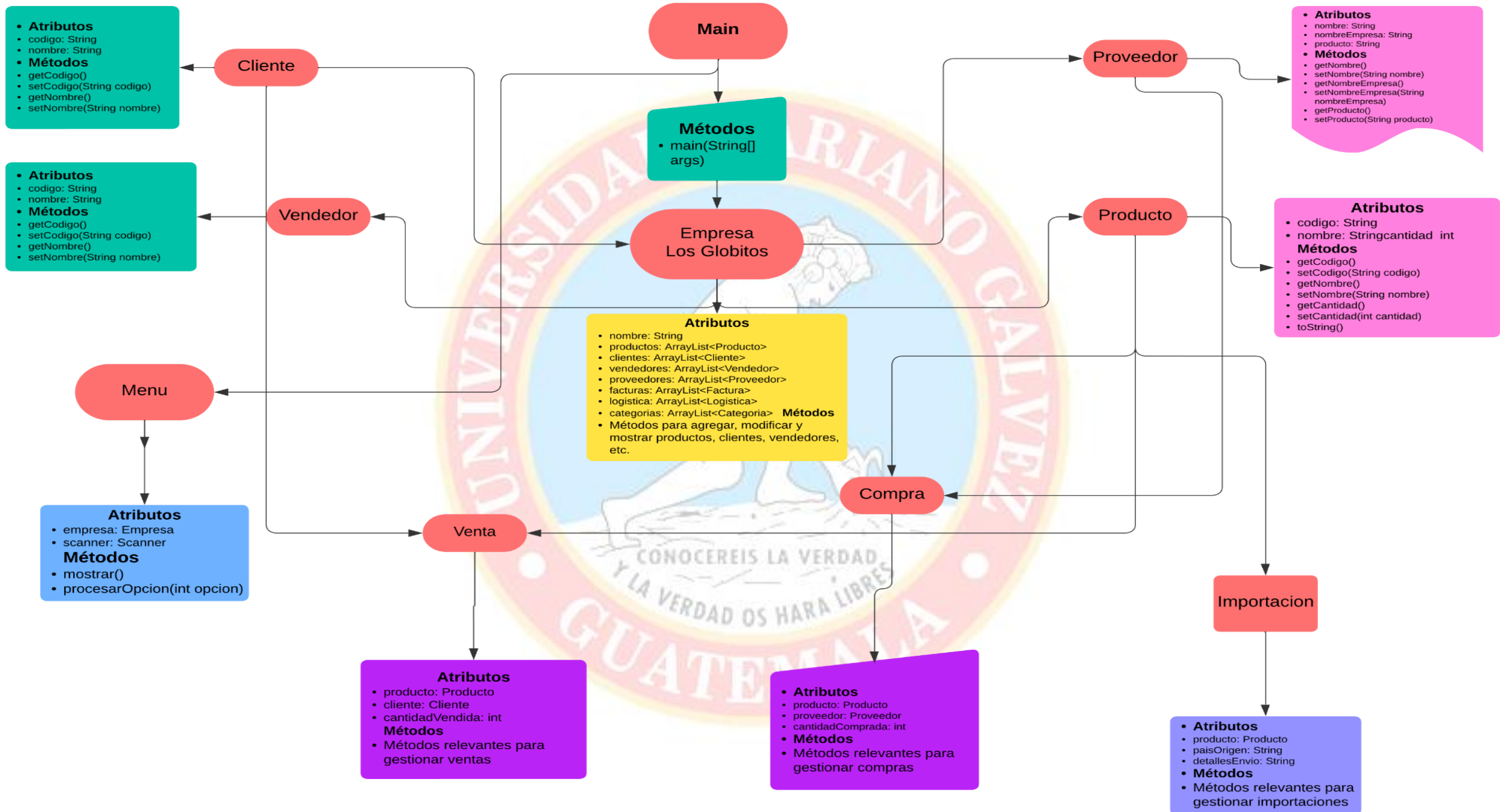
- **Atributos:**
  - codigo: String - Código del vendedor.
  - nombre: String - Nombre del vendedor.
- **Métodos:**
  - getCodigo(), setCodigo(String codigo): Métodos para obtener y establecer el código del vendedor.
  - getNombre(), setNombre(String nombre): Métodos para obtener y establecer el nombre del vendedor.

## Instrucciones de Uso

1. **Compilación:**
  - Navega al directorio donde se encuentran los archivos .java.
  - Ejecuta el comando: javac \*.java.
2. **Ejecución:**
  - Ejecuta el comando: java Main.

## Requisitos del Sistema

- **Java Development Kit (JDK):** Asegúrate de tener JDK instalado para compilar y ejecutar la aplicación.



## Relaciones Entre Clases

- **Empresa** tiene relaciones de composición con:
  - **Producto**
  - **Cliente**
  - **Vendedor**
  - **Proveedor**
  - **Factura**
  - **Logística**
  - **Categoría**
- **Venta** y **Compra** tienen una relación de asociación con **Producto**.
- **Venta** también tiene una relación de asociación con **Cliente**.
- **Compra** tiene una relación de asociación con **Proveedor**.
- **Importación** tiene una relación de asociación con **Producto**.

