



Universidad Mariano Gálvez de Guatemala  
Facultad de Ingeniería  
Ingeniería en Sistemas  
Curso: programación II

**TAREA:**

Documentación Proyecto Final

Autor: Wilmer Stuardo Jiménez García  
Carné: 7690-22-9822  
Guatemala, noviembre de 2023

## Tabla de Contenida

Introducción.....	3
Documentación .....	4
Introducción Al Proyecto. ....	4
Diagrama de Clases.....	4
Clases.....	4
Patrones de Diseño.....	5
Instrucciones de Uso.....	6
Requisitos del Sistema.....	6
Conclusión.....	10

## **Introducción**

Este proyecto tiene como objetivo brindar una solución integral para la administración de una juguetería, permitiendo a los propietarios y empleados gestionar eficazmente su inventario, ventas, pedidos y más.

En este proyecto, hemos creado clases y módulos que representan elementos clave de una juguetería, como los juguetes en stock, las ventas realizadas, los pedidos de productos y la interacción con una base de datos MariaDB para garantizar un almacenamiento eficiente y seguro de la información.

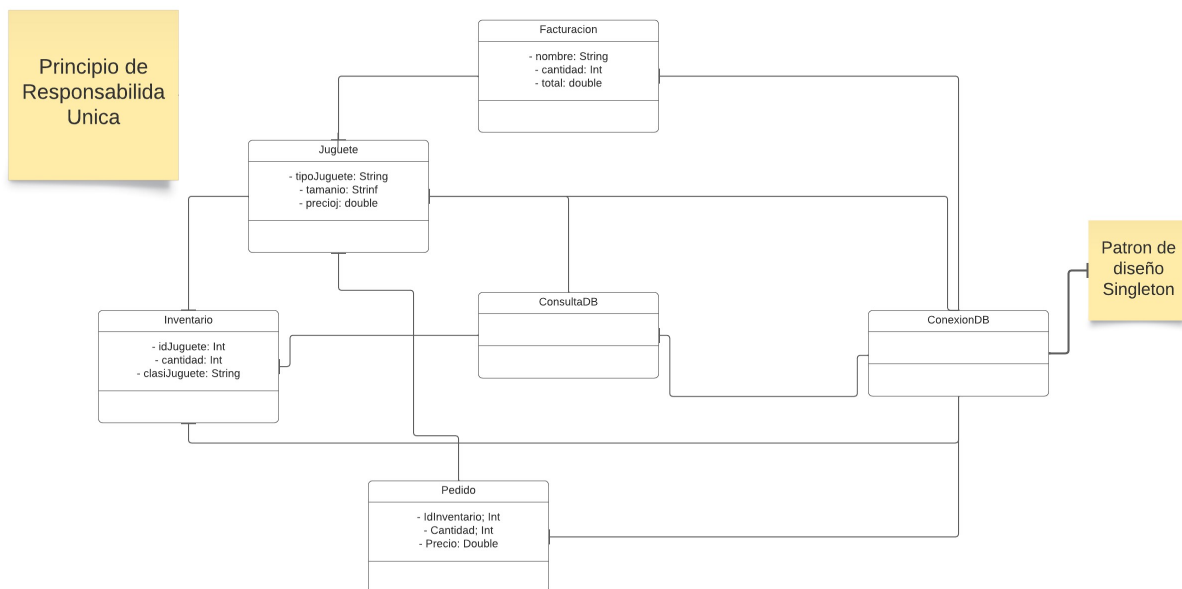
A lo largo de este proyecto, presentaremos las clases y métodos esenciales que componen este Sistema de Gestión de Juguetería, abordando las responsabilidades únicas de cada componente y cómo interactúan entre sí. Además, hemos creado un diagrama de clases UML que representa visualmente la estructura del sistema, lo que facilita la comprensión de su arquitectura.

## Documentación

### Introducción Al Proyecto.

El proyecto de sistema de gestión de juguetería es una aplicación de software diseñada para ayudar a administrar y controlar las operaciones de una tienda de juguetes. El software proporciona un conjunto de herramientas que permiten a los usuarios gestionar el inventario de juguetes, realizar ventas, realizar pedidos de reposición y mantener registros precisos de las transacciones comerciales.

### Diagrama de Clases



### Clases.

1. **Clase ConsultaDB**: Esta clase se encarga de interactuar con la base de datos. Proporciona métodos para consultar datos relacionados con los juguetes, el inventario y los pedidos. También puede actualizar y obtener información específica de la base de datos.
2. **Clase ConexionDB**: Esta clase se encarga de gestionar la conexión a la base de datos. Establece la conexión con la base de datos y proporciona métodos para ejecutar consultas SQL. Establece y administra la conexión a la base de datos.

3. **Clase Facturacion:** Esta clase se encarga de gestionar las ventas de juguetes. Permite a los usuarios comprar juguetes y actualiza el inventario y la información de facturación en la base de datos.
4. **Clase Jugue:** Esta clase permite la gestión de juguetes, incluyendo la adición de nuevos juguetes y la eliminación de juguetes existentes.
5. **Clase Pedido:** Esta clase se encarga de la gestión de pedidos, permitiendo a los usuarios realizar pedidos y eliminar pedidos existentes.
6. **Clase Jugueteria:** Esta clase actúa como la interfaz principal de usuario y permite a los usuarios interactuar con las funcionalidades de la tienda de juguetes. Muestra un menú de opciones al usuario para gestionar la interacción del usuario, permitiéndole realizar consultas, ventas, pedidos y otras operaciones.

### ***Patrones de Diseño***

- **Patrón Singleton:** El patrón Singleton se utiliza en la clase ConexionDB. Este patrón garantiza que solo exista una única instancia de una clase en toda la aplicación y proporciona un punto centralizado para acceder a esa instancia. Garantiza que todas las partes de la aplicación compartan la misma conexión a la base de datos, lo que evita problemas de consistencia y asegura que los datos se actualicen de manera coherente.
- **Principio de Responsabilidad Única:** El principio de Responsabilidad Única se aplica en todas las clases del proyecto. Este principio sugiere que cada clase debe tener una única razón para cambiar, es decir, una única responsabilidad. Cada clase en el presente proyecto se encarga de una funcionalidad específica y no asume múltiples responsabilidades. Por ejemplo, la clase Facturacion se encarga de la facturación y la gestión del inventario, la clase ConsultaDB se ocupa de la interacción con la base de datos, y así sucesivamente. Facilita la comprensión y el mantenimiento del código, ya que cada clase se enfoca en una tarea específica y no está sobrecargada con múltiples responsabilidades.

## Instrucciones de Uso

### ***Requisitos del Sistema***

- Java 8 o superior instalado
- Servidor MariaDB configurado y en funcionamiento

El programa brinda 7 tareas distintas que ayudara a la buena administración de la juguetería, entre las 7 opciones se encuentra:

```
Favor seleccione la opcion que desee
1. Consultar inventario
2. Catalogo
3. Vender Juguete
4. Realizar pedido
5. Eliminar Pedido
6. Anadir Juguete Nuevo
7. Eliminar Juguete
0. Salir
```

1. Consultar Inventario: Al seleccionar esta opción, mostrara todos los productos y la cantidad existente.

```
1
Inventario de Jugueteria:
juguete: Pelota - clasificacion: Deportes - cantidad: 57
juguete: muneca - clasificacion: muneca - cantidad: 25
juguete: Rompecabezas - clasificacion: Rompecabezas - cantidad: 20
juguete: Lego - clasificacion: Construccion - cantidad: 100
juguete: Carro de juguete - clasificacion: Vehiculos - cantidad: 65
juguete: trompo - clasificacion: nino - cantidad: 0
```

2. Catalogo: Al seleccionar esta opción, mostrara las características de cada uno de los productos que ofrece la tienda

```
2
Catalogo de Juguetes
id: 1 - Juguete: Pelota - tamaño: Pequeña - precio: 175.0
id: 2 - Juguete: muñeca - tamaño: Mediana - precio: 300.0
id: 3 - Juguete: Rompecabezas - tamaño: Grande - precio: 79.99
id: 4 - Juguete: Lego - tamaño: Pequeño - precio: 125.0
id: 5 - Juguete: Carro de juguete - tamaño: Mediano - precio: 95.5
id: 11 - Juguete: trompo - tamaño: pequeño - precio: 75.0
```

3. Vender Juguete: Al seleccionar esta opción se le permitirá al usuario realizar la venta de uno o más del producto existente

```
3
Lista de juguetes disponibles:
id: 1 - Juguete: Pelota - tamaño: Pequeña - precio: 175.0
id: 2 - Juguete: muñeca - tamaño: Mediana - precio: 300.0
id: 3 - Juguete: Rompecabezas - tamaño: Grande - precio: 79.99
id: 4 - Juguete: Lego - tamaño: Pequeño - precio: 125.0
id: 5 - Juguete: Carro de juguete - tamaño: Mediano - precio: 95.5
id: 11 - Juguete: trompo - tamaño: pequeño - precio: 75.0
Ingrese el ID del juguete que desea comprar: 2
Cantidad a comprar: 10
|*****|

Venta realizada con exito.

|*****|
```

4. Realizar Pedido: Al seleccionar esta opción el usuario podrá adquirir solicitar N cantidad de un producto en específico, esta opción esta pensada para cuando un producto este agotado o por agotarse.

```
4
----->LISTADO DE JUGUETES<-----
id: 1 - Juguete: Pelota - tamaño: Pequeña - precio: 175.0
id: 2 - Juguete: muñeca - tamaño: Mediana - precio: 300.0
id: 3 - Juguete: Rompecabezas - tamaño: Grande - precio: 79.99
id: 4 - Juguete: Lego - tamaño: Pequeño - precio: 125.0
id: 5 - Juguete: Carro de juguete - tamaño: Mediano - precio: 95.5
id: 11 - Juguete: trompo - tamaño: pequeño - precio: 75.0
Ingrese el ID del juguete que desea pedir: 5
Cantidad a pedir: 25
|*****|

Pedido realizado con exito.

|*****|
```

5. Eliminar Pedido: al seleccionar esta opción podrá eliminar algún pedido que se haya realizado con anterioridad únicamente eliminará la cantidad de producto que se había solicitado al momento de generar el pedido.

```
5
----->LISTADO DE JUGUETES DISPONIBLES<-----
id: 2 - Jugete: 2 - tamaño: 3 - precio: 25.5
id: 3 - Jugete: 3 - tamaño: 2 - precio: 35.0
id: 4 - Jugete: 4 - tamaño: 10 - precio: 22.75
id: 5 - Jugete: 5 - tamaño: 4 - precio: 28.25
id: 10 - Jugete: 11 - tamaño: 100 - precio: 7500.0
id: 14 - Jugete: 5 - tamaño: 25 - precio: 2387.5
Ingrese el ID del Pedido que desea eliminar: 14
|*****|

Eliminacion de Pedido realizado con exito.      I

|*****|
```

6. Añadir Jugete Nuevo: En esta opción se tendrá la oportunidad de añadir al inventario algún juguete que recientemente haya ingresado, es decir juguetes de temporadas o edición especial.

```
6
Ingrese el nombre del Jugete:
Barrilete
Ingrese el tamaño del nuevo Jugete:
mediano
Ingrese el precio:
35
Ingrese la clasificación del Jugete:
aereo
Nuevo juguete agregado con éxito.
```



7. Eliminar Juguete: esta opción permitirá eliminar cualquier juguete que ya no se desee vender en la tienda o que finalizo la temporada.

```
7
-----> LISTADO DE JUGUETES DISPONIBLES <-----
id: 1 - Juguete: Pelota - tamaño: Pequeña - precio: 175.0
id: 2 - Juguete: muñeca - tamaño: Mediana - precio: 300.0
id: 3 - Juguete: Rompecabezas - tamaño: Grande - precio: 79.99
id: 4 - Juguete: Lego - tamaño: Pequeño - precio: 125.0
id: 5 - Juguete: Carro de juguete - tamaño: Mediano - precio: 95.5
id: 11 - Juguete: trompo - tamaño: pequeño - precio: 75.0
id: 13 - Juguete: Barrilete - tamaño: mediano - precio: 35.0
Ingrese el ID del Juguete que desea eliminar: 13
|*****|

Eliminación del juguete realizada con éxito.

|*****|
```

0. Salir: Esta opción finalizara el programa, sin embargo, se puede estar tranquilo que los datos no se alteraran o se eliminaran debido a que todos estos datos se almacenaran en una base de datos.

```
0
Saliendo...
BUILD SUCCESSFUL (total time: 16 minutes 47 seconds)
```

jugueteria		jugueteriadb.juguete: 6 filas en total (a) >> Siguietes			
>	information_s...				
✓	jugueteriadb	112.0 KiB			
	facturacion	32.0 KiB			
	inventario	32.0 KiB			
	juguete	16.0 KiB			
	pedido	32.0 KiB			
	mysql				

idJuguete	tipoJuguete	tamaño	precio
1	Pelota	Pequeña	175.0
2	muñeca	Mediana	300.0
3	Rompecabezas	Grande	79.99
4	Lego	Pequeño	125.0
5	Carro de juguete	Mediano	95.5
11	trompo	pequeno	75.0

## **Conclusión**

En este proyecto de gestión de una tienda de juguetes, se ha logrado desarrollar un sistema que permite a los usuarios consultar el inventario, comprar juguetes, realizar pedidos y administrar el catálogo de productos. Para lograrlo, se aplicaron importantes principios de diseño y patrones de arquitectura de software que hacen que la aplicación sea eficiente, modular y fácil de mantener.

La arquitectura de este proyecto demuestra una buena práctica de diseño de software, lo que facilita su escalabilidad y extensión en el futuro. Además, la aplicación proporciona a los usuarios una interfaz intuitiva y funcional para gestionar productos y ventas en una tienda de juguetes.

este proyecto ilustra cómo los principios de diseño de software y los patrones arquitectónicos pueden contribuir a la creación de aplicaciones robustas y fáciles de mantener. Ofrece un ejemplo sólido de cómo aplicar el patrón Singleton y el principio de Responsabilidad Única en un contexto del mundo real, lo que beneficia tanto a desarrolladores como a usuarios finales.