

Universidad de las Fuerzas Armadas - ESPE Pregrado SII OCT24 – MAR25

Programación Orientada a Objetos - Nrc: 1323

Control de Lectura 2: Creación de Objetos y UML

Nombre de los integrantes:

David Iván Granada Pachacama Fernando Salvador Rodríguez Magallanes

Nombre del docente:

Mgtr. Luis Enrique Jaramillo Montaño

Fecha de entrega: 08 de diciembre del 2024



Contenido

Introducción			
1.	Objetivos	4	
1.1	. Objetivos General	4	
1.2	Objetivos Específicos	4	
2.	Marco teórico	4	
Componentes 5			
3.	Resultados Diagramas de clase UML	6	
4.	Resumen de resolución	8	
Análisis de las relaciones			
Resumen de Funcionalidades			
5.	Código en JAVA Apache NetBeans	9	
Interfaces9			
Clases con Implementación de Interfaces10			
6.	Conclusiones1	5	
7.	Recomendaciones1	5	
8.	Referencias Bibliográficas10	6	
9.	Apéndice10	6	



Tabla de figuras

Figura 1	6
Figura 2.	7
Figura 3.	9
Figura 4.	9
Figura 5.	10
Figura 6.	11
Figura 7.	11
Figura 8.	12
Figura 9.	12
Figura 10.	13
Figura 11	13
Figura 12.	14
Figura 13.	14
Figura 14	15



Introducción

En este informe se presenta el desarrollo e implementación de un sistema de gestión de talleres de vehículos utilizando la programación orientada a objetos (POO). El objetivo principal es diseñar un sistema que permita registrar talleres, gestionar vehículos, registrar ventas y consultar información clave de manera eficiente. El sistema es modular, escalable y puede ser aplicado a la gestión de talleres de diferentes tipos.

1. Objetivos

1.1. Objetivos General

 Desarrollar un sistema de gestión de talleres vehiculares que facilite el registro de talleres, la gestión de vehículos, la administración de ventas y la consulta de información relevante para optimizar las operaciones.

1.2. Objetivos Específicos

- Diseñar un sistema basado en POO que permita registrar talleres y vehículos asociados.
- Implementar funcionalidades para registrar y consultar ventas realizadas por cada taller.
- Proporcionar herramientas para visualizar el historial de ventas y actualizar resultados de manera eficiente.

2. Marco teórico

La programación orientada a objetos (POO) es un paradigma de programación que permite organizar el software en torno a objetos que interactúan entre sí. Este enfoque facilita la reutilización de código, el modularidad y el mantenimiento del sistema (Meyer, 1997).

Conceptos clave utilizados en este proyecto incluyen:

• Clases y Objetos: Representan entidades del sistema (e.g., Taller, Vehículo) y se definen como la base de la programación orientada a objetos. Las clases contienen atributos



y métodos que describen el comportamiento de los objetos que se crean a partir de ellas (Keene, 1989).

- Encapsulación: Protege los atributos internos de las clases mediante métodos públicos, permitiendo que las modificaciones internas de una clase no afecten el resto del sistema. Esto es clave para el mantenimiento y escalabilidad del código (Universitat Oberta de Catalunya, s.f.).
- Composición: Relación entre clases donde una clase contiene instancias de otra, indicando una relación de "todo-parte". Por ejemplo, un taller tiene una lista de vehículos; esta relación asegura que los vehículos no puedan existir sin el taller (Cachero Castro, 2013).
- Agregación: Relación más flexible entre clases, como la relación entre talleres y ventas, donde un objeto puede existir independientemente del otro. Este enfoque permite un diseño más modular y reutilizable (Fontela & Paez, 2015).

La implementación de estos conceptos en el sistema presentado permite estructurar las relaciones entre talleres, vehículos y ventas, facilitando la gestión eficiente y ampliando las posibilidades de personalización y escalabilidad.

Componentes

Un diagrama de clases cuenta con: el nombre, los atributos, las operaciones o comportamientos o métodos y las relaciones, elementos que se describen brevemente a continuación.

- Nombre: sirve de identificador para la clase, cada una de estas debe tener un nombre que no se repita
- Atributos: son las propiedades y características que tienen las clases, se los representa con el nombre del objeto, seguido del tipo de dato que es el atributo.
- Operaciones: también conocidos como métodos en Programación Orientada a Objetos,
 son las acciones que pueden ser realizadas por las instancias de esa clase y que



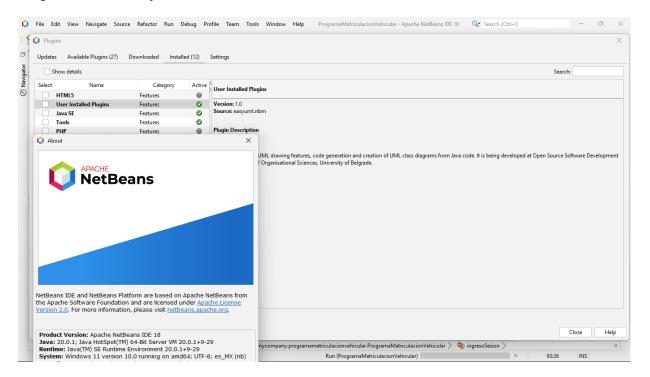
representa el comportamiento que tendrá dicha clase, se representan con: el nombre de la operación o método, los parámetros que tendrá y el tipo de retorno.

3. Resultados Diagramas de clase UML

A continuación, podemos observar en la Figura 1. el plug-in instalado para crear diagramas UML en NetBeans.

Figura 1.

Plugin instalado easyUML instalado en Windows 11

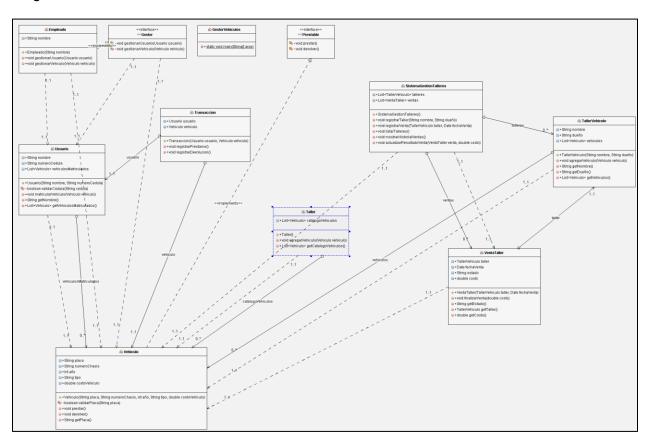




Posteriormente, podemos observar en la Figura 2. la creación de un ejemplo de un diagrama de clases empleando el plug-in easyUML llamado GestorVehiculos.

Figura 2.

Diagrama de clase GestorVehiculos UML





4. Resumen de resolución

- Vehículo: Representa los vehículos disponibles en un Taller, con atributos como placa, numero de chasis, año, evaluo, tipo, costo de matrícula, y métodos como prestar() y devolver().
- Usuario: Los usuarios pueden tener una lista de vehículos matriculados. Se relaciona con Vehículo mediante composición.
- Taller: Administra un catálogo de vehículos y se relaciona con Vehículo por asociación.
- Empleado: Gestiona usuarios y Vehículos, y se relaciona con Usuario y Vehículo por asociación.
- Transacción: Registra préstamos y devoluciones. Se relaciona con Usuario y Vehículo por agregación.

Análisis de las relaciones

- Composición: Usuario tiene una lista de Vehiculo. Si se elimina un Usuario, se eliminan sus vehículos matriculados.
- **Agregación:** Transaccion tiene referencias a Usuario y Vehiculo. Las transacciones pueden existir sin eliminar los usuarios o vehículos.
- Asociación: Taller tiene una lista de Vehiculo. Los vehículos pueden existir independientemente del taller.

Resumen de Funcionalidades

- Registrar Taller de Vehículo: Se puede registrar un taller con un nombre y un dueño, y agregar vehículos a ese taller.
- Crear Ventas de Talleres: Se registra una venta para un taller con la fecha de la venta.
- Consultar Información:
- Listar todos los talleres registrados con sus respectivos vehículos.



- Mostrar el historial de ventas de talleres de vehículos, incluyendo los talleres y los resultados.
- Actualizar Resultados de la Venta: Cambiar el estado de una venta de "Pendiente" a
 "Finalizado", añadiendo el costo.

5. Código en JAVA Apache NetBeans

Interfaces

Primero, definamos algunas interfaces que representen las acciones que pueden realizar las clases como se puede observar en las siguientes figuras.

Figura 3.

Creación de la Interfaz Prestable

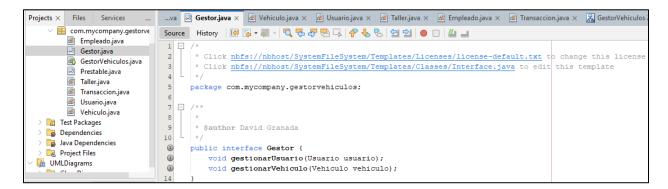


Figura 4.

Creación de la Interfaz Gestor

```
...va 🗵 Prestable,java × 🗵 Gestor,Java × 🙆 Vehiculo,java × 🙆 Usuario,java × 🖄 Taller,java × 🕸 Empleado,java × 🚳 Transaccion,java × 👪 GestorVehiculos 2...
Projects × Files Services __
       com.mycompany.gestorve
                                     com.mycompany.geston
Empleado.java
Gestor.java
Gestor.yehiculos.java
Prestable.java
Taller.java
Taller.java
Usuario.java
Vehiculo.java
                                             * Click nbfs://nbhost/SystemFileSystem/Templates/Licenses/license-default.txt to change this license
* Click nbfs://nbhost/SystemFileSystem/Templates/Classes/Interface.java to edit this template
                                      package com.mycompany.gestorvehiculos;
                                     8
9
10
    Test Packages
                                             * @author David Granada
    Dependencies
    Java Dependencies
                                            public interface Prestable {
     Project Files
                                                  void prestar();
 UMLDiagrams
                                                  void devolver();
```



Clases con Implementación de Interfaces

Ahora, adaptaremos las clases para que implementen estas interfaces como se puede observar en las Figuras 10 a la 16.

Figura 5.

Creación Clase Vehiculo

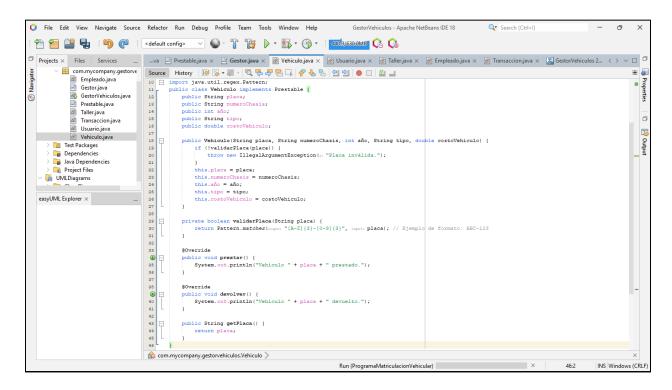




Figura 6.

Creación Clase Usuario (Composición)

```
O File Edit View Navigate Source Refactor Run Debug Profile Team Tools Window Help
                                                                                                                                                                                                                                                                     GestorVehiculos - Apache NetBeans IDE 18
    The transfer of the transfer o
 Projects x Files Services _ ...va 🖆 Prestablejava x 🖆 Gestor/Java x 🧸 destor/Java x 🧸 destor/Java x 🧸 destor/Java x 🔻 destor/Java x 🖟 Busuariojava x 🖟 Tallerjava x destor/Java x 🖟 Transaccion.java x 🛣 Gestor/Vehiculos 2... < > v 🗆 🗅
                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                      Properties
                      Empleado.java
Gestor.java
                                                                                                  11 import java.util.ArrayList;
                                  Gestor, Java
Gestor, Jeva
Prestable, Java
Taller, Java
Transaccion, Java
                                                                                                import java.util.List;
import java.util.regex.Pattern;
                                                                                               public class Usuario (
public String nombre;
public String nombre;
public String nombre;
public String numeroCedula;
public List'eWhiculo' vehiculosMatriculados; // Compo

public Usuario(String nombre, String numeroCedula) {
    if (fvalidarCedula(combininumeroCedula)) {
        throw new TllegalArgumentException(s: "Cédula )
}
}

■ Usuario.java

                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                             Untput 🖳
                                        Vehiculo.java
              > Test Packages
               > 📴 Dependencies
> 📴 Java Dependencies
                                                                                                                                        if (!validarCedula(codula:numeroCedula)) {
   throw new IllegalArgumentException(s: "Cédula inválida.");
                      Project Files
             UMLDiagrams
                                                                                                                                         this.nombre = nombre;
                                                                                                                                       this.numeroCedula = numeroCedula;
        easyUML Explorer ×
                                                                                                                                      this.vehiculosMatriculados = new ArrayList<>();
                                                                                                                                                                                                                      "[0-9]{10}", input: cedula); // Ejemplo de formato: 1234567890
                                                                                                                            public void matricularVehiculo(Vehiculo vehiculo) {
                                                                                                                                      vehiculosMatriculados.add(@:vehiculo);
System.out.println("Vehiculo " + vehiculo.getPlaca() + " matriculado por " + nombre);
                                                                                                                            public String getNombre() {
                                                                                                                             public List<Vehiculo> getVehiculosMatriculados() {
                                                                                                  숨 com.mycompany.gestorvehiculos.Usuario >
                                                                                                                                                                                                                                                                   Run (ProgramaMatriculacionVehicular)
                                                                                                                                                                                                                                                                                                                                                                                                                                                                                         INS Windows (CRLF)
```

Figura 7.

Creación Clase Taller (Asociación)

```
🔾 File Edit View Navigate Source Refactor Run Debug Profile Team Tools Window Help
                                                                                                                                                                                                                                      GestorVehiculos - Apache NetBeans IDE 18
   ** The state of th
Projects x Files Services __ ...va 🗟 Prestablejava x 🗟 Gestor/java x 🗗 Gestor/java x 🗗 Vehiculo.java x 🗗 Usuario.java x 🛣 Taller.java x 🛣 Empleado.java x 🚳 Transaccion.java x 🖫 Gestor/Vehiculos 2... < > > 🗆
        Properties
                                                                                                   package com.mycompany.gestorvehiculos;
                             Transaccion.jav
Usuario.java
Vehiculo.java
                                                                                                     * @author David Granada
                                                                                                                                                                                                                                                                                                                                                                                                                                                                1
                                                                                                                                                                                                                                                                                                                                                                                                                                                                Output
            > 🧰 Test Packages
                                                                                     11 import java.util.ArrayList;
12 import java.util.List;
             > 📴 Dependencies
               Java Dependencies
          ☐ UMLDiagrams
                                                                                                                public List<Vehiculo> catalogoVehiculos: // Asociación
                                                                                       17 public Taller() {
        easyUML Explorer ×
                                                                                       18
                                                                                                                           this.catalogoVehiculos = new ArrayList<>();
                                                                                       21 public void agregarVehiculo (Vehiculo vehiculo) {
                                                                                                                catalogoveniculos.adu(e: veniculo;,
System.out.println("Vehiculo " + vehiculo.getPlaca() + " agregado al taller.");
                                                                                       24
                                                                                                               public List<Vehiculo> getCatalogoVehiculos() {
                                                                                       28
29
                                                                                                                                                                                                                                   Run (ProgramaMatriculacionVehicular)
                                                                                                                                                                                                                                                                                                                                                                                                                               INS Windows (CRLF
```



Figura 8.

Creación Clase Empleado (Asociación)

```
O File Edit View Navigate Source Refactor Run Debug Profile Team Tools Window Help
                                                                                                                                                                                                                         GestorVehiculos - Apache NetBeans IDE 18
   The latest two figs and the latest two figs are latest two figs and the latest two figs are latest two figs. The latest two figs are latest two figs are latest two figs. The latest two figs are latest two figs are latest two figs. The latest two figs are latest two figs are latest two figs. The latest two figs are latest two figs are latest two figs. The latest two figs are latest two figs are latest two figs. The latest two figs are latest two figs are latest two figs are latest two figs. The latest two figs are latest two figs are latest two figs are latest two figs are latest two figs. The latest two figs are latest two figs. The latest two figs are latest two figs are latest two figs are latest two figs. The latest two figs are latest two figs are latest two figs are latest two figs are latest two figs. The latest two figs are latest two figs are latest two figs. The latest two figs are latest two figs are latest two figs are latest two figs. The latest two figs are latest two figs are latest two figs are latest two figs. The latest two figs are latest two figs are latest two figs are latest two figs are latest two figs. The latest two figs are latest two figs are latest two figs are latest two figs are latest two figs. The latest two figs are latest two figs. The latest two figs are latest two figs a
 Projects x Files Services _ ...va 🖆 Prestablejava x 🖆 Gestor/Java x 🤞 Vehiculo.java x 🔞 Usuario.java x 🔞 Tallerjava x 🔞 Empleado.java x 🔞 Transaccion.java x 🖫 Gestor/Vehiculos 2... < > v 🗆 🗅
                                                                                                                                                                                                                                                                                                                                                                                                                                Properties
                      * Click nbfs://nbhost/SystemFileSystem/Templates/Licenses/license-default.txt to change this license
                            Gestor/Vehiculos.java
Prestable.java
Taller.java
Transaccion.java
                                                                                                 * Click nbfs://nbhost/SystemFileSystem/Templates/Classes/Class.java to edit this template
                                                                                             package com.mycompany.gestorvehiculos;
                                                                                                                                                                                                                                                                                                                                                                                                                                       6

    Usuario.java
    Vehiculo.java

                                                                                                                                                                                                                                                                                                                                                                                                                                       T.
                                 Vehiculo.java
                                                                                           * @author David Granada
            > ear Test Packages
             Dependencies
Java Dependencies
                                                                                             public class Empleado implements Gestor {
                  Project Files
                                                                                                         public String nombre;
           UMLDiagrams
                                                                                  public Empleado (String nombre) {
                                                                                         this.nombre = nombre;
      easyUML Explorer ×
                                                                                                                              void gestionarUsuario(Usuario usuario) {
                                                                                                         System.out.println("Gestionando usuario: " + usuario.getNombre());
}
                                                                                  20
21
22
                                                                                                          23
                                                                                    ⓐ 📮
                                                                                  25
26
                                                                                                                                                                                                                      Run (ProgramaMatriculacionVehicular)
                                                                                                                                                                                                                                                                                                                                                                                                       INS Windows (CRLF)
```

Figura 9.

Clase Transaccion (Agregación)

```
File Edit View Navigate Source Refactor Run Debug Profile Team Tools Window Help
                                                                                                                                                                                                                                             GestorVehiculos - Apache NetBeans IDE 18
                                                                                                                                                                                                                                                                                                                                                    Search (Ctrl+I)
   The second secon
🗗 Projects x Files Services _ ...va 🕒 Prestable_java x 🖃 Gestor_java x 🦝 Gestor_java x 🏕 Usuario_java x 🏕 Taller_java x 🛣 Empleado_java x 🏚 Transaccion_java x 🛣 Gransaccion_java x
                                                                                    Source History | № 🕞 • 🗐 • | 🥰 💆 📑 | 🔗 😓 | № 💇 • □ | 🕌 🚊
                                                                                                                                                                                                                                                                                                                                                                                                                                                                  Properties
                              Empleado.java
Gestor.java
                                                                                                         * Click nbfs://nbhost/SystemFileSystem/Templates/Licenses/license-default.txt to change this license
* Click nbfs://nbhost/SystemFileSystem/Templates/Classes/Class.java to edit this template
                              Gestor/ehiculos.java
Prestable.java
Taller.java
                                                                                         5 pac)
6
7 - /**
                                                                                                      package com.mycompany.gestorvehiculos;
                                                                                                                                                                                                                                                                                                                                                                                                                                                                        6
                              Usuario.javaVehiculo.java
                                                                                                                                                                                                                                                                                                                                                                                                                                                                        Ī.
                                                                                                   * @author David Granada
             > 🧰 Test Packages
               Dependencies
                 Java Dependencies
                                                                                                     public class Transaccion {
                                                                                                         public Usuario usuario; // Agregación
public Vehiculo vehiculo; // Agregación
           ☐ UMLDiagrams
        easyUML Explorer ×
                                                                                        15 public Transaccion(Usuario usuario, Vehiculo vehiculo) {
                                                                                                                             this.usuario = usuario:
                                                                                                                  this.vehiculo = vehiculo;
                                                                                                               public void registrarPrestamo() {
                                                                                                                                            culo.prestar();
                                                                                                                       System.out.println("Préstamo registrado para " + usuario.getNombre());
                                                                                        25 📮
                                                                                                                  public void registrarDevolucion() {
                                                                                                                          vehiculo.devolver();
System.out.println("Devolución registrada para " + usuario.getNombre());
                                                                                        29
30
                                                                                                                                                                                                                                       Run (ProgramaMatriculacionVehicular)
                                                                                                                                                                                                                                                                                                                                                                                                                                      INS Windows (CRLF
```



Figura 10.

Clase Taller Vehiculo

```
🔾 File Edit View Navigate Source Refactor Run Debug Profile Team Tools Window Help
                                                                                                                                                                                                                                                                                                                                        GestorVehiculos - Apache NetBeans IDE 18
                                                                                                                                                                                                                                                                                                                                                                                                                                                                                  Search (Ctrl+I)
                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                               a
     ** The state of th
  🗗 Projects x Files Services _ ...va 🗟 Transaccion.java x 📓 Gestor/Vehiculos 2024.12.08 21-00-07.cdg x 🙋 Taller/Vehiculojava x 🖄 VentaTaller.java x 🐼 SistemaGestionTalleres.java x
                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                          v 🗆 🗇
GestorVehiculos
Source Packag
com.mycor
                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                  Properties
                                                                                                                        Source Packages

a com.mycompany.gestorvehicu

b g
                               Empleado.java
Gestor.java
GestorVehiculos.java
Prestable.java
                                                                                                                       10 */
11 = import java.util.ArrayList;
12 import java.util.List;
13
14 public class TallerVehiculo
15 public String nombre;
16 public String dueño;
17 public List-Vehiculo> ve
                                                                                                                                          public class TallerVehiculo {
                               SistemaGestionTalleres.java
Taller.java
                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                          U Output
                                                                                                                                              public String nombre;
public String dueño;
public List<Vehiculo> vehiculos; // Composición

    TallerVehiculo.java

                               Transaccion.java

Usuario.java
                                                                                                                         public TallerVehiculo(String nombre, String dueño) {
    this.nombre = nombre;
    this.dueño = dueño;
    this.vehiculos = new ArrayList<);
}

public void agregarVehiculo(Vehiculo vehiculo) {
    vehiculos.add(u.vehiculo);
    System.out.println("Wehiculo" + vehiculo.getPlaca() + " agregado al taller " + nombre);
}

public String getNombre() {
    return nombre;
}

public String getNombre() {
    return nombre;
}

public String getNombre() {
    return nombre;
}

public String getNombre() {

■ Vehiculo.iava

                               VentaTaller.java
          easyUML Explorer ×
                                                                                                                                                        public String getDueño() {
                                                                                                                          com.mycompany.gestorvehiculos.TallerVehiculo
                                                                                                                                                                                                                                                                                                                                   Run (ProgramaMatriculacionVehicular)
                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                            INS Windows (CRLF)
```

Figura 11.

Clase Venta Taller

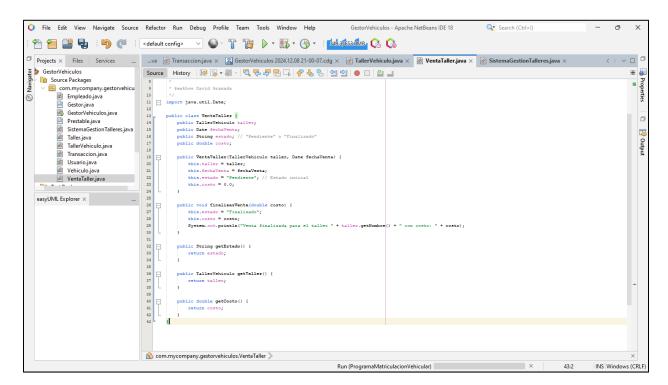




Figura 12.

Clase SistemaGestionTalleres

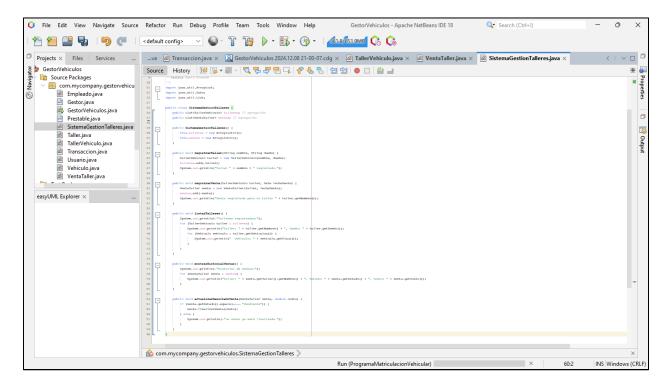


Figura 13.

Creación Clase Principal GestorVehicular

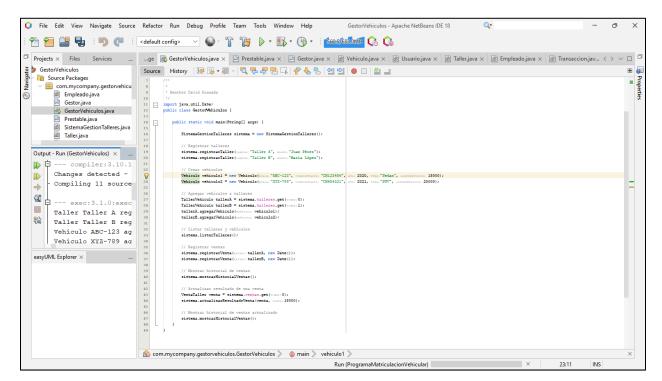
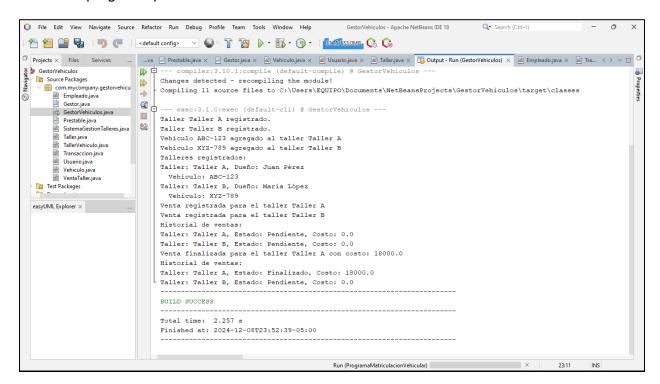




Figura 14.

Salida del programa por consola



6. Conclusiones

- El sistema desarrollado cumple con los objetivos planteados al permitir la gestión eficiente de talleres, vehículos y ventas.
- La utilización de POO facilitó la modularidad y escalabilidad del sistema, permitiendo futuras expansiones.
- El diseño UML permitió visualizar las relaciones entre clases y guiar la implementación.

7. Recomendaciones

- Incorporar una base de datos para almacenar de forma persistente la información de talleres, vehículos y ventas.
- Implementar una interfaz gráfica para facilitar la interacción con el usuario.
- Ampliar las funcionalidades para incluir reportes avanzados y gestión de clientes.



8. Referencias Bibliográficas

- Cachero Castro, C. (2013). Introducción a la programación orientada a objetos. Textos docentes. Recuperado de
 - https://sedici.unlp.edu.ar/bitstream/handle/10915/29797/Clase_2013.pdf?sequence=3
- Fontela, C., & Paez, N. (2015). Programación Orientada a Objetos. Recuperado de https://www.academia.edu/11537928/Libro programacion orientada a objetos POO
- Keene, S. E. (1989). Object-Oriented Programming in Common Lisp: A Programmer's Guide to CLOS. Addison-Wesley. https://doi.org/10.5555/534810
- Meyer, B. (1997). Object-Oriented Software Construction (2^a ed.). Prentice Hall. https://doi.org/10.5555/261119
- Universitat Oberta de Catalunya. (s.f.). Introducción al paradigma de la programación orientada a objetos. Recuperado de

 https://openaccess.uoc.edu/bitstream/10609/149901/2/IntroduccionAlParadigmaDeLaPr ogramacionOrientadaAObjetos.pdf

9. Apéndice

A continuación, adjunto link GitHub de mi repositorio donde se encuentra la actividad Con los archivos completos: https://github.com/digranada/202451 1323.git