

Laboratorio de
programación III

Proyecto final:

Gestión de inventario de concesionaria
automotriz

Integrantes:

FUENTES IGNACIO, LABATUT MARIANO

Profesor:

CHIESSA LEONARDO

Ciclo Lectivo:

2024

Índice:

Introducción	2
Propósito Principal	2
Usuarios Destinatarios	2
Tecnologías Utilizadas	2
Arquitectura del Proyecto	2
Estructuras de las clases principales empleadas.....	3
Diagrama UML.....	4
Interfaz de Usuario.....	4
Bibliografía.	8

Índice de ilustraciones

<u>Ilustración 1: Interfaz gráfica de inicio de sesión</u>	4
<u>Ilustración 2: Interfaz gráfica principal</u>	5
<u>Ilustración 3: Menú para agregar vehículos</u>	5
<u>Ilustración 4: Menú de modificación de vehículos</u>	6
<u>Ilustración 5: Menú de eliminación de vehículos</u>	6
<u>Ilustración 6: Interfaz principal luego de actualizar la tabla de vehículos</u>	7
<u>Ilustración 7: Interfaz de búsqueda</u>	7
<u>Ilustración 8: Interfaz principal luego de filtrar los vehículos de marca Ford</u>	8

Introducción

En el sector automotriz, la gestión eficiente del inventario es crucial para el éxito de una concesionaria. La correcta administración de vehículos, tanto autos como motocicletas, permite no solo optimizar los procesos internos, sino también mejorar la experiencia del cliente, incrementar las ventas y reducir costos operativos. El sector automotriz, fundamental en la economía global, requiere herramientas avanzadas que faciliten la gestión y el control de inventarios para mantenerse competitivo y rentable.

Propósito Principal

El propósito fundamental de esta aplicación es proporcionar una solución integral y eficiente para la administración de inventarios en concesionarias de vehículos. Diseñada para ofrecer un control riguroso y preciso del stock, esta aplicación permite a las concesionarias mantener un registro detallado y actualizado de sus vehículos, facilitando la toma de decisiones estratégicas y operativas.

Usuarios Destinatarios

Esta herramienta está dirigida principalmente a concesionarias de autos y motos que buscan mejorar su eficiencia operativa y administrativa. En particular, la aplicación ha sido desarrollada pensando en satisfacer las necesidades de "MotorClass", una concesionaria ubicada en Mar del Plata que requiere una herramienta robusta para gestionar su inventario de manera efectiva y segura.

Tecnologías Utilizadas

El desarrollo de esta aplicación se ha llevado a cabo utilizando el lenguaje de programación Java, aprovechando la robustez y versatilidad que este ofrece. La interfaz gráfica ha sido implementada con Swing, permitiendo una experiencia de usuario interactiva y amigable. Para la persistencia de datos, se ha optado por el uso de archivos en formato JSON, evitando la necesidad de un sistema de gestión de bases de datos y facilitando la portabilidad de la información.

La organización del equipo de desarrollo se realizó mediante Trello, lo que permitió una gestión ágil y eficiente de las tareas.

Adicionalmente, para el diseño y documentación de la arquitectura del sistema, se utilizó Lucidchart, herramienta que facilitó la creación de diagramas UML claros y precisos.

Con estas premisas, el presente documento detallará la arquitectura, funcionalidades y procedimientos asociados a este sistema, brindando una herramienta sólida y efectiva para la gestión de inventarios en el contexto específico de concesionarias de vehículos.

Arquitectura del Proyecto

Estructuras de las clases principales empleadas

• **Vehículo**

La clase abstracta Vehículo constituye la base sobre la cual se derivan otros tipos específicos de vehículos, como autos y motos. Esta clase contiene los atributos esenciales que describen cualquier vehículo gestionado en la concesionaria. A continuación, se detallan los atributos de la clase Vehículo:

Tipo de dato	Nombre	Variable que almacena
Int	id	Identificador único del vehículo.
String	marca	Marca del vehículo (por ejemplo, Toyota, Honda).
String	modelo	Modelo específico del vehículo (por ejemplo, Corolla, Civic).
String	color	Color del vehículo.
Int	año	Año de fabricación del vehículo.
Double	precioCosto	Precio de adquisición del vehículo por parte de la concesionaria.
Int	stock	Cantidad disponible de este modelo de vehículo en el inventario.

Tabla 1: Detalle de cada atributo en la clase Vehículo.

• **Automóvil**

La clase automovil, al heredar de la clase Vehículo posee los mismos atributos, pero además se le agrega el siguiente.

Tipo de dato	Nombre	Variable que almacena
Int	cantPuertas	Cantidad de puertas que posee el automovil
TipoAutomovil	TipoAutomovil	Sus opciones son limitadas y se listan en un ENUM.

Tabla 2: Detalle de cada variable en la clase Automóvil.

• **Moto**

La clase Moto, al heredar de la clase Vehículo posee los mismos atributos, pero además se le agrega el siguiente.

Tipo de dato	Nombre	Variable que almacena
Double	Cilindrada	Capacidad de cilindrada que posee la moto

Tabla 3: Detalle de cada variable en la clase Moto.

Diagrama UML

Se anexa un diagrama UML en el repositorio mostrando las distintas clases e interfaces y como se relacionan entre si.

•Información Consolidada y Archivos

El proyecto cuenta con dos archivos en formato json.

El archivo con el nombre "usuarios.json" almacena las identidades de los usuarios con sus contraseñas, para luego poder hacer las validaciones y darle acceso a la aplicación. Se puede escalar a otros usuarios, con la posibilidad de agregar distintos roles.

El archivo con el nombre "vehículos.json" almacena la colección que contiene todos los vehículos del inventario de la concesionaria. Este archivo es el que se modifica constantemente al crear, modificar o eliminar los vehículos.

Interfaz de Usuario

El usuario al iniciar el programa se encuentra con una ventana de logeo, donde puede ingresar sus credenciales.

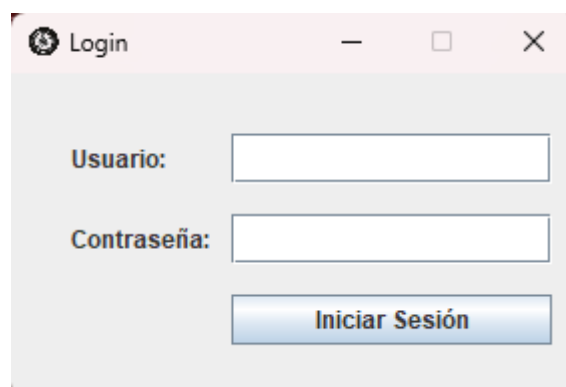


Ilustración 1: Interfaz gráfica de inicio de sesión.

Luego, si las credenciales coinciden con las almacenadas en el archivo, se muestra la pantalla principal de la aplicación.

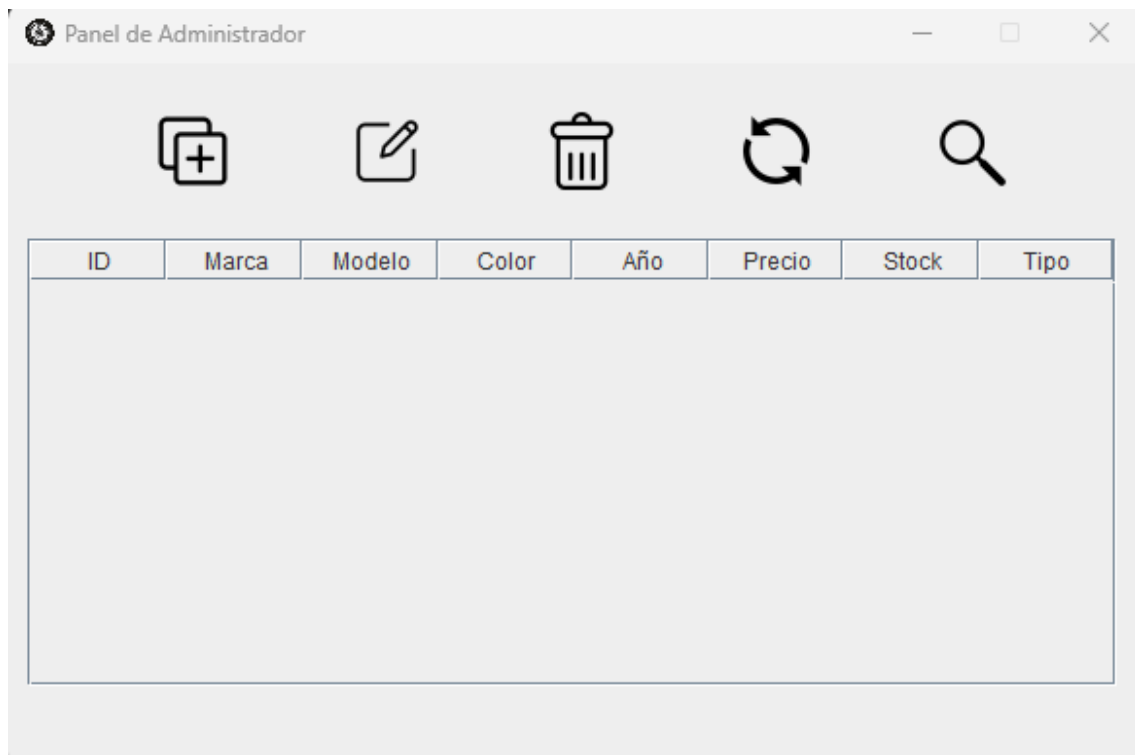


Ilustración 2: Interfaz gráfica principal

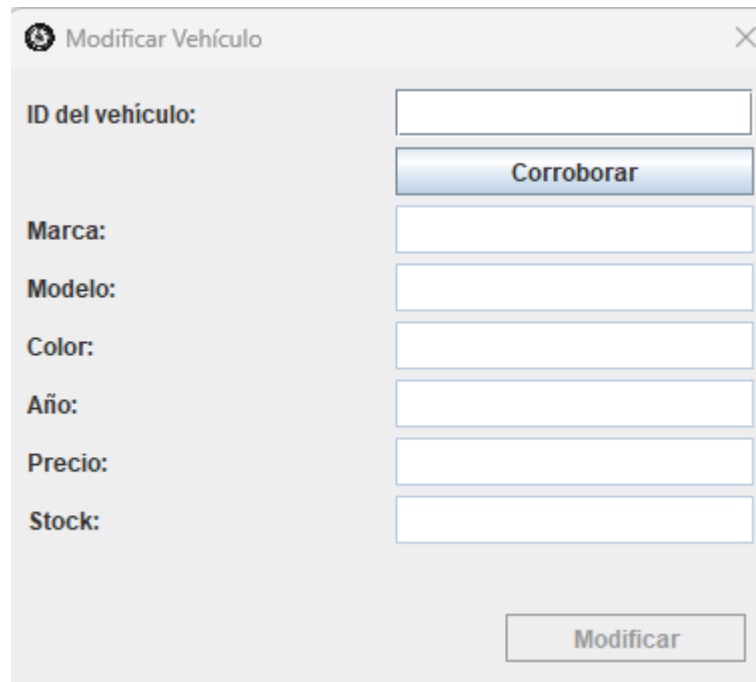
En la parte superior se encuentran los cinco botones principales y, en la parte inferior, una tabla destinada a mostrar los datos de los vehículos solicitados.

El primer botón permite crear nuevos vehículos, desplegando la siguiente interfaz gráfica:

Ilustración 3: Menú para agregar vehículos

Es necesario ingresar las distintas características del vehículo, las cuales incluyen: la marca, el modelo, el color, el año, el precio, la cantidad en stock y el tipo (automóvil o moto). En el caso del año, no puede ser inferior a 2000, ya que la concesionaria no admite modelos antiguos.

El segundo botón se utiliza para modificar los vehículos ya agregados.



Modificar Vehículo

ID del vehículo:

Corroborar

Marca:

Modelo:

Color:

Año:

Precio:

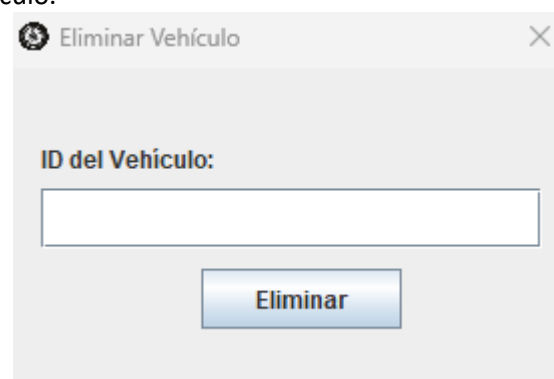
Stock:

Modificar

Ilustración 4: Menú de modificación de vehículos

Para modificar un vehículo, es necesario ingresar el ID de la unidad. Al presionar "Corroborar", se cargan los datos del vehículo o se muestra un mensaje indicando que el vehículo no existe. Si el vehículo se encuentra, se habilitan los campos para modificar, los cuales son los mismos que se solicitan al crear un nuevo vehículo.

El tercer botón se utiliza para eliminar vehículos. Este proceso también se realiza mediante el ID del vehículo.



Eliminar Vehículo

ID del Vehículo:

Eliminar

Ilustración 5: Menú de eliminación de vehículos

Si el vehículo está en el archivo, se pide confirmación si realmente se desea eliminar. Si el vehículo se encuentra, se avisa mediante un mensaje.

El cuarto botón se utiliza para cargar todos los vehículos almacenados en stock. También se utiliza para actualizar la lista de vehículos que se muestran.



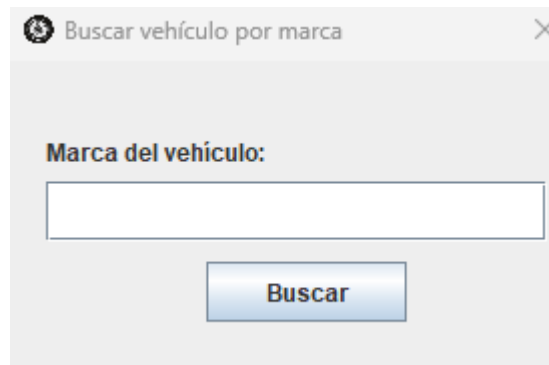
Panel de Administrador

Icons: Add (+), Edit (pencil), Delete (trash), Refresh (circular arrow), Search (magnifying glass).

ID	Marca	Modelo	Color	Año	Precio	Stock	Tipo
6	BMW	X5	Azul	2019	60000.0	21	automovil
7	Audi	A4	Negro	2021	45000.0	4	automovil
8	Mercedes-...	C-Class	Blanco	2020	55000.0	7	automovil
9	Hyundai	Elantra	Gris	2019	21000.0	9	automovil
10	Volkswagen	Jetta	Negro	2021	28000.0	5	automovil
11	Toyota	Rav4	Gris	2022	30000.0	6	automovil
12	Honda	Accord	Rojo	2020	27000.0	4	automovil
13	Ford	Mustang	Azul	2021	45000.0	3	automovil
14	Yamaha	YZF-R3	Azul	2020	5500.0	10	moto
15	Kawasaki	Ninja 400	Verde	2019	6000.0	8	moto

Ilustración 6: Interfaz principal luego de actualizar la tabla de vehículos.

El ultimo botón se utiliza para buscar vehículos por marca.



Buscar vehículo por marca

Marca del vehículo:

Buscar

Ilustración 7: Interfaz de búsqueda.

Al encontrar el vehículo, la interfaz principal mostrara solamente los vehículos que posean la marca buscada.

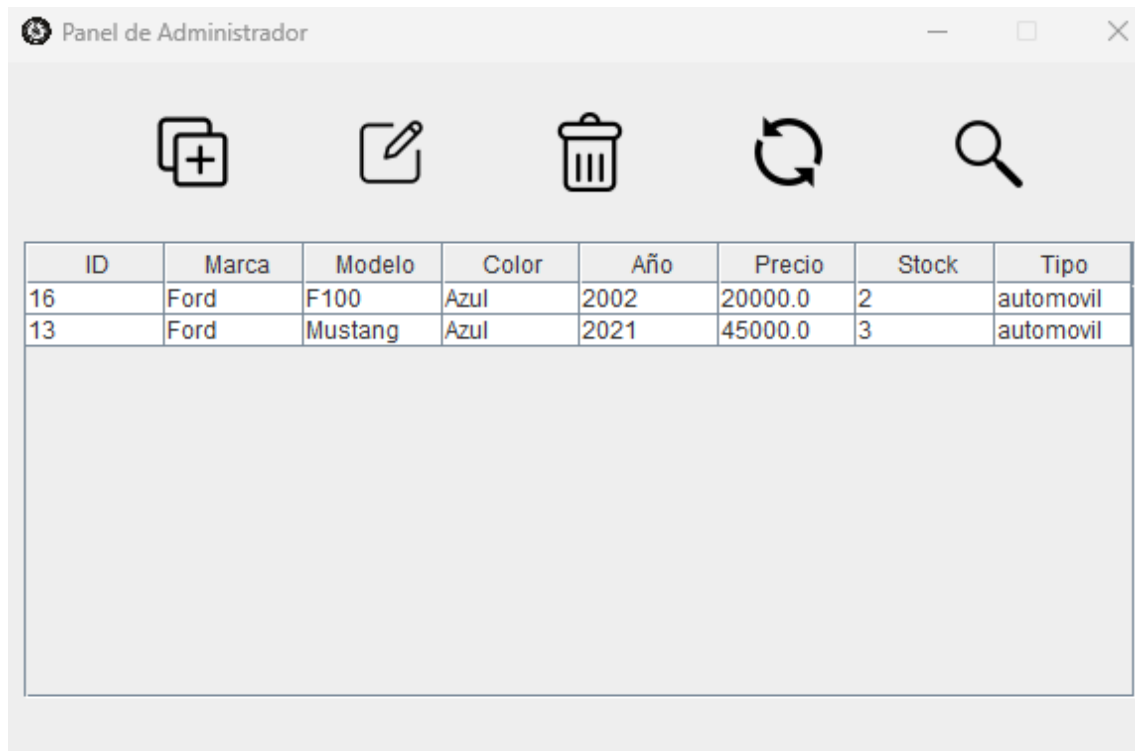


Ilustración 8: Interfaz principal luego de filtrar los vehículos de marca Ford

Bibliografía.

- Joyanes, L. (2011). PROGRAMACION EN JAVA 6.
- Documentación oficial de Oracle:
JDK 22 documentation. (2024, abril 1). Oracle Help Center.
<https://docs.oracle.com/en/java/javase/22/>
- Clases teóricas proporcionadas por la cátedra