

GA4-220501095-AA2-EV04

Diagrama de Clases del proyecto de software

Aprendices

Adriana Gineth Barrios Aponte

Juan Carlos Cadena Muñoz

Servicio Nacional de Aprendizaje SENA

Análisis y desarrollo de software ADSO

Ficha 3118560

Instructor: Carlos Ernesto Maya Vallejo

CENTRO DE FORMACIÓN

comercio y servicios

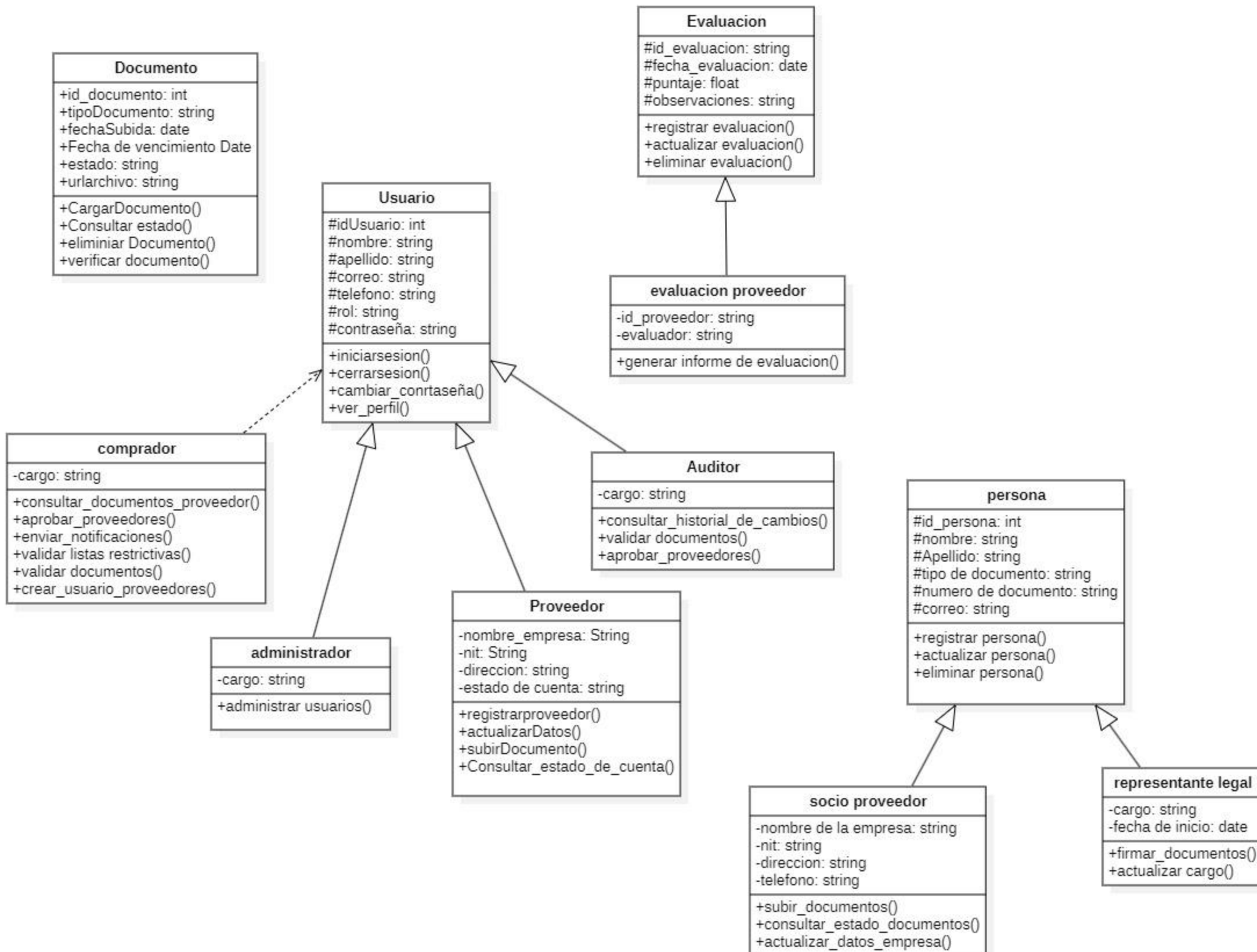
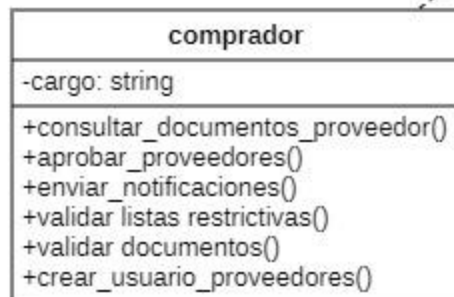
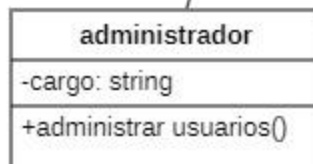
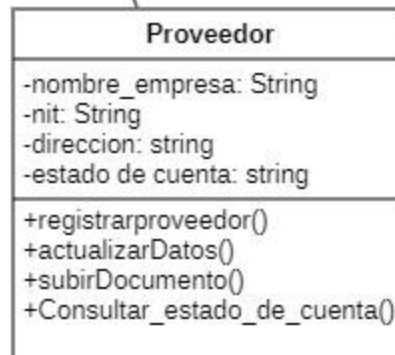
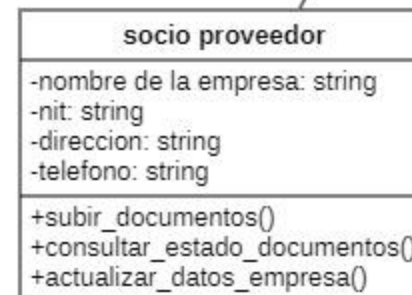
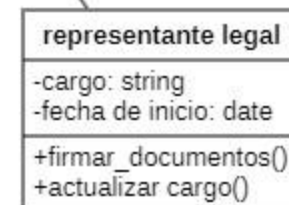
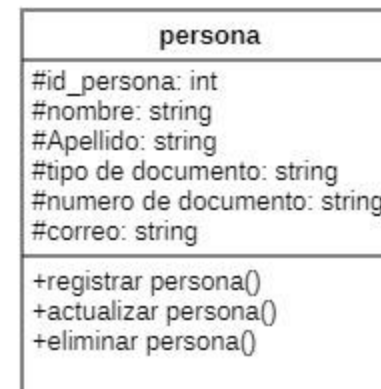
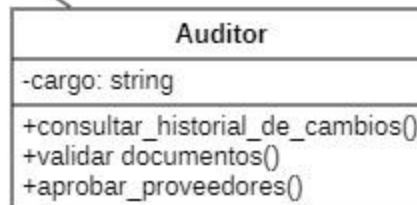
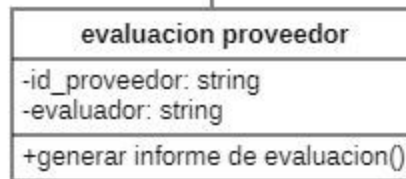
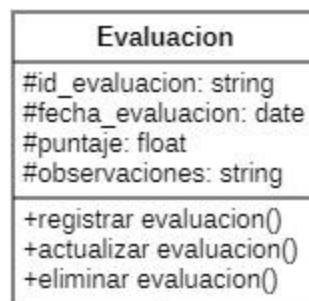
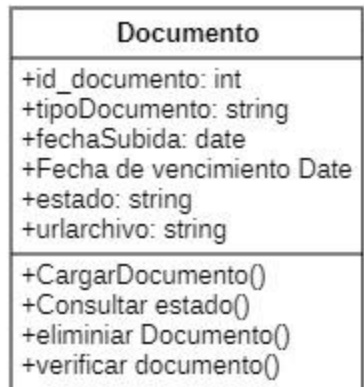
Popayán – Cauca, 19 de agosto de 2025

INTRODUCCIÓN

La programación orientada a objetos (POO) permite modelar sistemas complejos de manera organizada mediante el uso de clases, atributos, métodos y relaciones de herencia. En el diseño del portal de proveedores de **Golden Odds**, se implementó un modelo basado en diagramas de clases que reflejan la estructura lógica del sistema y la interacción entre sus componentes principales.

El modelo parte de clases generales, como **Usuario**, **Persona**, **Documento** y **Evaluación**, que concentran atributos y métodos comunes para posteriormente ser heredados por clases más específicas. Este enfoque evita la duplicidad de código, mejora la reutilización y facilita el mantenimiento del software.

El diagrama de clases desarrollado constituye una representación conceptual que muestra cómo se gestionarán los usuarios, proveedores, documentos y evaluaciones dentro del sistema, garantizando así claridad en la definición de responsabilidades y robustez en la arquitectura de la aplicación.



CONCLUSIONES

- La construcción de los diagramas de clases permitió identificar claramente las **entidades clave del sistema**, sus atributos y comportamientos, así como las relaciones de herencia necesarias para aprovechar las ventajas de la POO.
- El uso de clases generales como **Usuario, Persona, Documento y Evaluación** asegura un diseño modular y escalable, ya que facilita que las clases hijas amplíen las funcionalidades sin necesidad de repetir código.
- La organización del modelo con **herencia** permite reflejar de forma fiel los roles y responsabilidades dentro del portal de proveedores, garantizando que cada actor (administrador, auditor, compras, proveedor) tenga sus operaciones específicas sin perder la coherencia del sistema.
- Este diseño ofrece una base sólida para la futura **implementación en código**, asegurando trazabilidad entre el análisis, el diseño y el desarrollo. Además, facilita la comunicación entre los miembros del equipo, ya que el diagrama actúa como un lenguaje común entre analistas, diseñadores y programadores.
- En conclusión, los diagramas de clases elaborados constituyen un paso fundamental dentro de la metodología de desarrollo, permitiendo que el sistema proyectado cumpla con principios de reutilización, claridad, transparencia y escalabilidad.