

Definir estándares de codificación de acuerdo a plataforma de desarrollo elegida GA7-  
220501096-AA1-EV02.

Danna Michel Gallegos Moreno

SENA

Análisis y desarrollo de software

2721448

Cristian Arias

Marzo 24, 2024

## INTRODUCCION

Abordar el modelo conceptual bajo el paradigma orientado a objetos, es importante ya que permite relacionar los conceptos con la práctica, estos conceptos le permiten al desarrollador crear nuevos modelos a partir del conocimiento de clase, objetos, métodos y atributos y de las herramientas case.

En el desarrollo de software se evidencia que el paradigma orientado a objetos facilita la realización de las tareas y conlleva a realizar un trabajo de calidad y confiabilidad para el usuario.

La programación orientada a objetos (POO) es un paradigma el cual se adapta a diversos contextos y a través de su código permite un trabajo ágil. En el encontramos:

- **Clase:** son elementos que contienen los objetos y que se relacionan unos a otros para programar.
- **Objetos:** son unidades que interactúan entre si y a los que se le atribuye atributos y métodos.
- **Atributos:** son características individuales y propiedades de un objeto.
- **Métodos:** describen el comportamiento del objeto y cuando se da la orden se realiza una serie de operaciones

## PRINCIPIO DE LA PROGRAMACIÓN ORIENTADA A OBJETOS

Guarda y ocupa parte de la información permitiendo garantizar los datos y solo expone la información elegida al exterior

**Abstracción:** simplifica lo complejo del sistema y permite al usuario interactuar con los atributos y métodos seleccionados de un objeto.

**Herencia:** son relaciones de clase que tienen algunas características y comportamientos similares

## CONTROL DE ACCESO

Muestra la accesibilidad al sistema y puede ser:

- 🚦 **Público:** con acceso por todas las clases
- 🚦 **Privado:** se da visibilidad o accesibilidad a ciertas clases
- 🚦 **Protegido:** se tiene acceso a el desde la propia clase que lo define y las que los hereda

## HERRAMIENTAS CASE

Son herramientas para automatizar las tareas del desarrollo del software, enlaza y automatiza las fases del ciclo de vida.

Entre ellas encontramos:

- ❖ **Herramientas integradas y case:** componen todas las fases del ciclo de vida del sistema.
- ❖ **Herramientas de alto nivel U-case:** Dirigidas a la automatización y soporte de las actividades en las fases de análisis y desarrollo.
- ❖ **Herramientas de bajo nivel L-case:** se dan e la construcción e implementación del sistema.
- ❖ **Juegos de herramientas:** son las simples automatizan las fases y están dirigidas al mantenimiento del software.

## CONCLUSIONES

- Se logra reconocer y aprender los términos básicos en la programación orientada a objetos.
- Es necesario implementar los procesos para construir un sistema de información